Burroughs

B 6700

SYSTEM SOFTWARE IMPROVEMENTS

D NOTE DOCUMENTATION

(RELATIVE TO MARK 2.6 RELEASE)

Printed in U.S. America

1 April 1974

5000763 (2.6)

TABLE OF CONTENTS

NEW FEATURES AND DOCUMENTATION CHANGES

ALGOL

D0551 ALGOL - DOLLAR CARDS - 12-06-73

THIS CHANGE IMPLEMENTS THE ABILITY TO SET DOLLAR OPTIONS EQUAL TO BOOLEAN EXPRESSIONS COMPOSED OF *, EQV, IMP, OR, AND, NOT, TRUE, FALSE AND USER OPTIONS:

```
$SET OMIT = OPT1 AND OPT2 OR NOT OPT3
$SET OMIT = * OR OPTO
```

D0583 ALGOL - ALGOL INTERFACE TO DMS II - 11-18-73

THIS CHANGE IMPLEMENTS THE ALGOL INTERFACE TO DMS II BURROUGHS DATA MANAGEMENT SYSTEM. DETAILED DOCUMENTATION OF THIS INTERFACE MAY BE FOUND IN THE B6700/B7700 DMSII HOST LANGUAGE INTERFACE (ALGOL AND COBOL) TO BURROUGHS DATA MANAGEMENT SYSTEM (5000839).

D0592 ALGOL - BATCH COMPILING - ALGOL - 12-06-73

INTRODUCTION

FREQUENTLY USERS FIND THAT THEY CAN GROUP A LARGE NUMBER OF PROGRAMS WITH THESE SIMILAR OUTWARD APPEARANCES:

1. THEY ARE COMPILE FOR SYNTAX, OR COMPILE AND GO JOBS.
2. IF THEY MAKE IT INTO EXECUTION, THE EXECUTION TIMES ARE LOW.

3. NO MORE THAN ONE PRINTER FILE AND NO MORE THAN ONE CARD-READER FILE IS USED.

4. NO OPERATOR INTERVENTION IS NEEDED TO RUN THE JOB.

5. IF TASKING IS AVAILABLE IN THE LANGUAGE BEING USED, IT IS NOT BEING USED IN THE PROGRAM.

CERTAINLY ONE CATEGORY OF JOBS THAT MATCHES THE ABOVE DESCRIPTION ARE "STUDENT JOBS", BUT THERE ARE ACTUALLY A NUMBER OF CLASSES OF OTHER KINDS OF JOBS THAT ALSO FIT THE ABOVE DESCRIPTION.

A CAREFUL EXAMINATION OF THE COMPUTER TIME NEEDED TO COMPILE SUCH JOBS WOULD SHOW THAT A SIZABLE PERCENTAGE (OFTEN WELL OVER 50%) OF THE ELAPSED AND PROCESSING TIMES IS SPENT PERFORMING INITIALIZATION AND TERMINATION FUNCTIONS: OPENING AND CLOSING FILES, ALLOCATING AND DEALLOCATING ARRAYS, BUILDING STACKS, LOGGING, LINKING INTO INTRINSICS, MAKING CODE SEGMENTS PRESENT, ETC. WHILE ALL OF THESE FUNCTIONS ARE LARGELY UNAVOIDABLE, IT WOULD BE DESIRABLE IF SOME PROGRAMS COULD BE MADE TO SHARE THE COST OF THESE INITIALIZATION AND TERMINATION PROCEDURES. ACHIEVING THIS GOAL IS THE IDEA BEHIND BATCH COMPILERS.

SEVERAL SCHEMES FOR BATCH COMPILERS HAVE BEEN SUGGESTED FOR THE B6700/B7700. THE FOLLOWING WAS CHOSEN:

DECK SET UP FOR A BATCH COMPILER

A DECK FOR A BATCH COMPILATION AND EXECUTION IS A GROUP OF "JOBS" ALL WRITTEN IN THE SAME LANGUAGE, EACH OBSERVING THE RESTRICTIONS IMPOSED ON THAT LANGUAGE BY BATCHING (SEE RESTRICTIONS AT THE END) AND EACH DELIMITED FROM THE OTHER BY THE FOLLOWING ORDER OF CARDS:

```
$JOB  OPTIONAL JOB TITLE
      PROGRAM

$ENTRY
DATA, IF ANY
```

THE FIRST CARD IN THE DECK IS A "$ JOB" CARD, WHICH IS SOLELY USED TO INDICATE THE BEGINNING OF THE PROGRAM. OPTIONALLY, A JOB TITLE

MAY APPEAR ON THE $ JOB CARD; THIS TITLE WILL BE USED TO IDENTIFY PRINTED OUTPUT PRODUCED BY THE COMPILER. FOLLOWING THE $ JOB CARD IS THE SOURCE PROGRAM, COMPLETE WITH THE NORMAL COLLECTION OF COMPILE $ CONTROL CARDS, ETC.

THE NEXT ITEM IS THE "$ ENTRY" CARD. THIS CARD IS USED TO INDICATE THE END OF THE PROGRAM. IT IS ALSO USED TO INDICATE THAT EXECUTION IS DESIRED IF THERE WERE NO ERRORS. IF THE "$ ENTRY" CARD IS MISSING, THE COMPILER ASSUMES THAT NO EXECUTION OF THIS PROGRAM WAS WANTED, I.E., ONLY A "COMPILE FOR SYNTAX" WAS REQUESTED.

FINALLY, IF THE PROGRAM USES A DATA DECK (I.E., THE PROGRAM REFERENCES A FILE WITH A "KIND" OF "READER") THEN THE DATA DECK APPEARS AFTER THE "$ ENTRY" CARD. NOTE THAT THE "$ ENTRY" CARD IS REQUIRED FOR EXECUTION REGARDLESS OF WHETHER OR NOT THERE IS A DATA DECK.

ADDITIONALLY, THERE ARE SOME NEW CONTROL CARD OPTIONS THAT APPLY TO THE CONTROL OF EXECUTION:

```
PROCESSTIME
IOTIME
```

THESE TWO OPTIONS ($ OPTIONS) ARE IDENTICAL TO THEIR MCP CONTROL CARD COUNTERPARTS OF THE SAME NAME, THAT IS, FOLLOWING THESE OPTIONS IS A TIME EXPRESSED IN SECONDS OR FRACTIONS OF SECONDS. FOR EXAMPLE:

```
$ PROCESSTIME = 2.2     IOTIME = 5
```

WOULD CAUSE AN UPPER LIMIT OF 2.2 SECONDS ON EXECUTION TIME AND FIVE SECONDS IN I/O TIME TO BE SET. THERE IS NO WAY TO LIMIT ELAPSED TIME, SINCE PROGRAMS CANNOT CONTROL THE ELAPSED TIME.

PROCESSTIME AND IOTIME MAY BE SET EITHER BEFORE THE FIRST "$ JOB" CARD OR AFTER ANY SUBSEQUENT "$ JOB" CARD. IF EITHER OF THESE OPTIONS ARE SET BEFORE THE FIRST "$ JOB" CARD THEN THE VALUE THEY ARE SET TO ACT AS UPPER BOUNDS FOR ANY PROCESSTIME AND IOTIME OPTIONS THAT MIGHT BE SET WITHIN THE INDIVIDUAL USER PROGRAMS. THE IDEA BEHIND THESE RESTRICTIONS IS THAT THE INSTALLATION CONTROLS

THE MAXIMUM TIMES INDIVIDUAL PROGRAMS RUN, BUT THE USER MAY ENFORCE LOWER TIMES FOR HIS OWN JOB.

AN EXAMPLE OF A BATCH OF THREE FORTRAN JOBS

```
<I>COMPILE BATCH FORTRAN;DATA
$PROCESSTIME = 5.5  IOTIME = 1

$JOB                    JOB # 1
$SET FREE RESET SINGLE
REAL A(10)
READ 1,A
PRINT 1,A
1     FORMAT(10A6)
END
$ENTRY
THIS IS A DATA CARD FOR THE ABOVE JOB

$JOB                     JOB # 2
         WRITE(6,1) (SIN(3.14*I),  I=1, 20)
1     FORMAT(1H , F13.4)
      I=X + + X        %SYNTAX ERROR
      END
$ENTRY

$JOB                    JOB # 3
C***          COMPILE FOR SYNTAX
      SUBROUTINE XXXXXX
      DO 10 I=1,10
10     CONTINUE     %A SILLY LOOP
      RETURN
      END
      X=Y
$SET CODE
      CALL XXXXXX
      STOP
      END       % THERE IS NO "$ ENTRY" CARD -- JUST A COMPILE
                     FOR SYNTAX
```

<I>END

NOTE THAT NO JOB IN THIS THREE-JOB BATCH WILL BE ALLOWED MORE THAN 5.5 SECONDS OF PROCESSOR TIME, NOR MORE THAN ONE SECOND OF I/O TIME. NOTE ALSO THAT JOB#1 WILL COMPILE CORRECTLY AND RUN, USING THE ONE CARD DATA DECK IMMEDIATELY FOLLOWING ITS $ ENTRY CARD; THAT JOB#2 HAS A SYNTAX ERROR, BUT WOULD HAVE RUN HAD IT BEEN ERROR FREE (IT HAS NO DATA DECK, HOWEVER); AND THAT JOB#3, WHILE BEING SYNTACTICALLY CORRECT, WILL NOT RUN BECAUSE IT HAS NO $ ENTRY CARD.

RESTRICTIONS

CURRENTLY THERE ARE ONLY TWO COMPILERS THAT HAVE BATCHING FACILITIES: FORTRAN AND ALGOL. SEVERAL RESTRICTIONS MUST BE MET BY A PROGRAM IF THE PROGRAM IS TO BE INCLUDED IN A "BATCH STREAM".

1. THE PRINTER FILE MAY NOT BE EXPLICITLY OPENED, CLOSED, OR HAVE ITS ATTRIBUTES CHANGED. ATTEMPTS TO DO THIS WILL TERMINATE THE "JOB".
2. BINDING IS NOT ALLOWED.
3. MISSING SUBROUTINES OR FUNCTIONS ARE FATAL ERRORS.
4. THE PAUSE (FORTRAN) AND WAIT (ALGOL) STATEMENTS ARE ALLOWED, BUT DO NOT STOP THE RUNNING OF A JOB.
5. ANY RSVP SITUATION, SUCH AS NO FILE, ETC., IS A FATAL EXECUTION ERROR.
6. ONLY ONE PRINTER FILE IS ALLOWED. IF TWO OR MORE FILES ARE DECLARED TO BE PRINTER, THEY WILL BE "JOINED" INTO ONE FILE. LIKEWISE, AT MOST ONE READER FILE IS ALLOWED.
7. TASKING IS DISALLOWED.

IMPLEMENTATION SCHEME

THE GOAL OF THE IMPLEMENTATION WAS TO ELIMINATE AS MUCH NORMAL SYSTEM OVERHEAD AS POSSIBLE BY REDUCING THE NUMBER OF TASKS INITIATED IN THE SYSTEM WITHIN THE NATURAL RUNNING ENVIRONMENT OF

THE B6700/B7700.

IN ORDER TO ELIMINATE MANY INITIATIONS OF A COMPILER, THE INDIVIDUAL JOBS ARE COLLECTED INTO A BATCH AND PRESENTED AS ONE FILE TO THE BATCHING COMPILER. THIS OBVIOUSLY REDUCES THE NUMBER OF COMPILES TO ONE COMPILE, ENABLING THE COMPILER TO "GET UP TO SPEED". THE COMPILATION PROCESS FOR EACH INDIVIDUAL JOB IS VIRTUALLY THE SAME AS FOR NON-BATCHED JOBS AND YIELDS EQUALLY EFFICIENT OBJECT CODE. WHEN THE COMPILER FINISHES COMPILING THE LAST INDIVIDUAL JOB, IT GENERATES SPECIAL OBJECT CODE IN THE OUTER BLOCK TO LINK EACH INDIVIDUAL JOB TO THE NEXT ONE. SHOULD ANY INDIVIDUAL JOB HAVE SYNTAX ERRORS, OR SPECIFY "COMPILE FOR SYNTAX" IT IS NOT LINKED INTO THE OTHER JOBS. THE CODE OF ALL INDIVIDUAL JOBS RESIDES IN ONE CODE FILE ON DISK AT THE END OF THE COMPILE.

THE PRINTED OUTPUT FROM THE COMPILER IS DIRECTED TO A BACKUP DISK FILE WITH AN ALTERED BDBASE SO THAT IT WILL NOT BE PRINTED BY AUTOPRINT. LOGGING INFORMATION REGARDING THE COMPILE IS ALSO SAVED IN THIS FILE. THE EXECTUION OF THE CODE IS:

1. BUILD THE D2 STACK
2. CALL BATCHMONITOR PASSING IT A PROCEDURE WHICH SERIALLY CALLS EACH INDIVIDUAL JOB.
3. BATCHMONITOR PROCESSES THE PROCEDURE PASSED TO IT. IF ANY JOB SHOULD CAUSE A FATAL EXECUTION ERROR, BATCHMONITOR REPROCESSES THE PROCEDURE, WHICH SEQUENCES AUTOMATICALLY TO THE NEXT INDIVIDUAL JOB.
4. BATCHMONITOR REWINDS THE TWO BACKUP PRINTER FILES, EXTRACTS THE LOGGING INFORMATION, AND COLLATES THE OUTPUT INTO A NEW PRINTER FILE.
5. RETURN TO THE MCP.

THE COMPILER MAKES ATTEMPTS TO SHARE ARRAYS FROM ONE INDIVIDUAL JOB WITH SUCCEEDING JOBS, ELIMINATING MANY PRESENCE BIT INTERRUPTS. ADDITIONALLY, ALL JOBS SHARE THE SAME PRINTER FILE AND INTRINSICS AND MAY EVEN SHARE THE SAME CODE SEGMENTS. THE INDIVIDUAL JOBS RUN

SERIALLY AND SHARE THE SAME STACK SPACE.

ONE JOB IS PROTECTED FROM PREVIOUS JOBS BY THE BATCHMONITOR, A DCALGOL INTRINSIC. SHOULD ONE JOB HAVE AN ERROR, THE EXECUTION IS REINITIATED BY BATCHMONITOR AT THE NEXT INDIVIDUAL JOB. SHOULD A JOB USE AN EXCESSIVE AMOUNT OF EITHER I/O TIME OR PROCESSOR TIME, THIS FACT IS NOTICED BY BATCHMONITOR, AND THE INDIVIDUAL JOB IS TERMINATED. LIKEWISE, BATCHMONITOR ENFORCES THE RULE THAT NO RSVP MESSAGES ARE ALLOWED, BY TERMINATING THE JOB THAT CAUSES ONE.

BATCHMONITOR EXTRACTS THE LOGGING INFORMATION FROM THE TWO PRINTER FILES AND SUMMARIZES IT AT THE END OF THE OUTPUT OF EACH INDIVIDUAL JOB. THIS IS EASILY MODIFIED TO INTERFACE WITH THE ACCOUNTING SYSTEM OF A GIVEN INSTALLATION. ADDITIONALLY, TWO WORDS IN EACH LOG RECORD FURNISHED BY THE COMPILER AND THE INDIVIDUAL JOB ARE SPARES TO FACILITATE ANY INSTALLATION EXTENSIONS.

D0643 ALGOL - CHECKPOINT FUNCTION IN ALGOL - 01-06-74

THIS CHANGE IMPLEMENTS A NEW ALGOL/DCALGOL/DMALGOL FUNCTION, CHECKPOINT. THE SYNTAX AND SEMANTICS ARE AS FOLLOWS:

SYNTAX:

<CHECKPOINT FUNCTION>::= CHECKPOINT(<DEVICE>,<DISPOSITION>)
<DEVICE>::= DISK / DISKPACK / PACK
<DISPOSITION>::= LOCK / PURGE

SEMANTICS:

THE <CHECKPOINT FUNCTION> (BOOLEAN VALUED) CAUSES A CHECKPOINT TO BE TAKEN WHENEVER IT OCCURS AS PART OF A BOOLEAN EXPRESSION, E.G., BOOL:=CHECKPOINT(DISK,LOCK) (PLEASE REFER TO THE SYSTEM MISCELLANEA FOR A DETAILED DISCUSSION OF THE CHECKPOINT OPERATION). THE VALUE RETURNED BY THE <CHECKPOINT FUNCTION> IS THE STATUS OF THE ACTUAL CHECKPOINT, OR THE STATUS OF THE RESTART. NOTE THAT THE NEW $ OPTION, CHECKRESTART (DESCRIBED BELOW), MUST BE SET AT THE START OF

THE PROGRAM IF THE PROGRAM IS TO CONTAIN CALLS ON THE <CHECKPOINT FUNCTION>.

NEW $ OPTION:

A NEW $ OPTION HAS BEEN ADDED TO HANDLE THE <CHECKPOINT FUNCTION>. IF A CHECKPOINT IS TO BE PERFORMED WITHIN THE PROGRAM, THE $ OPTION "CHECKRESTART" SHOULD BE SET AT THE START OF THE PROGRAM. CHECKRESTART IS RESET BY DEFAULT.

D0658 ALGOL - TRANSLATE ATTRIBUTES - 12-28-73

THIS PATCH ADDS FOUR NEW FILE ATTRIBUTES, ALL PERTAINING TO SOFTWARE TRANSLATION. THE FOLLOWING ATTRIBUTES AND MNEMONICS SHOULD BE ADDED TO SECTION 6.2 OF THE ALGOL MANUAL.

TO <FILE ATTRIBUTE IDENTIFIER> ADD:

```
<TRANSLATE-TABLE-VALUED FILE ATTRIBUTE NAME>
<TRANSLATE-TABLE-VALUED FILE ATTRIBUTE NAME>::= INPUTTABLE /
                                                OUTPUTTABLE
```

TO <ARITHMETIC-FILE ATTRIBUTE NAME> ADD:

```
TRANSLATE
```

TO <BOOLEAN-VALUED FILE ATTRIBUTE NAME> ADD:

```
TRANSLATING
```

TO <MNEMONIC FILE ATTRIBUTE VALUE> ADD:

```
<TRANSLATE MNEMONIC>
<TRANSLATE MNEMONIC>::= DEFAULTTRANS / FULLTRANS / SOFTONLY /
                        NOSOFT / NOTRANS / FORCESOFT
```

FOR A COMPLETE EXPLANATION OF THE SOFTWARE TRANSLATION ATTRIBUTES, PLEASE REFER TO MCP-I-O D0594.

D0659 ALGOL - CTMON DOLLAR OPTION - 01-20-74

D0659 ALGOL - CTMON DOLLAR OPTION - 01-20-74

THIS CHANGE EXTENDS THE MEANING OF THE CTMON DOLLAR OPTION TO PRINT ON THE LISTING THE CURRENT VALUE OF A COMPILE-TIME VARIABLE WHEN IT IS REFERENCED, AS WELL AS THE NEW VALUE WHEN IT IS CHANGED.

D0682 ALGOL - USER SEGMENTATION - 10-28-73

THIS CHANGE ALLOWS BLOCKS, AS WELL AS PROCEDURES, TO BE CONTAINED IN USER SEGMENTS. PRIOR TO THIS CHANGE ONLY PROCEDURES WERE PERMITTED.

D0696 ALGOL - LISTOMITTED DOLLAR OPTION - 10-28-73

THIS CHANGE IMPLEMENTS THE LISTOMITTED DOLLAR CARD OPTION. THIS OPTION, SET BY DEFAULT, MUST BE EXPLICITLY RESET. WHEN RESET, SOURCE CODE CARDS BEING OMITTED WILL NOT BE PRINTED OUT ON LISTINGS. THE SETTING AND POPPING OF DOLLAR CARDS WILL, HOWEVER, BE PRINTED IF THE $LIST OPTION IS SET.

THIS OPTION IS DESIGNED TO AID IN FOLLOWING PROGRAM LOGIC WHERE MANY COMBINATIONS OF OMITTING ARE FREQUENTLY USED.

D0697 ALGOL - LINENUMBER ADDED TO ALGOL - 10-28-73

THE FEATURE LINENUMBER THAT WAS FORMERLY IN DMALGOL IS NOW IN ALGOL WITH A SLIGHT CHANGE. FORMERLY LINENUMBER PRODUCED THE OLD SEQUENCE NUMBER, BUT NOW LINENUMBER GIVES THE NEW SEQUENCE NUMBER, I.E., THE RESULTANT NUMBER AFTER MERGING OR RESEQUENCING. WHEN THIS FEATURE IS USED WITH DMALGOL THE NEW SEQUENCE NUMBER WILL BE GIVEN, AND IN SOME CASES THIS MAY BE DIFFERENT THAN WHAT WAS PREVIOUSLY PRODUCED BY DMALGOL. FOR EXAMPLE, THIS FEATURE MAY BE INVOKED BY X:=LINENUMBER WHERE X IS DECLARED AS INTEGER X. THE

D0697 ALGOL - LINENUMBER ADDED TO ALGOL - 10-28-73

EXECUTION OF THIS STATEMENT CAUSES THE CURRENT SEQUENCE NUMBER OF THE CARD BEING READ TO BE STORED IN X.

D0707 ALGOL - ALGOL RUN STATEMENT - 02-10-74

PROCEDURES INVOKED VIA THE RUN STATEMENT (I.E., AS AN INDEPENDENT TASK) MUST EITHER HAVE NO PARAMETERS OR HAVE ALL PARAMETERS DECLARED VALUE. A SYNTAX ERROR IS NOW GIVEN IF A PROCEDURE WITH A CALL-BY-NAME PARAMETER IS INVOKED VIA THE RUN STATEMENT. NOTE THAT ARRAYS OR FILES CANNOT BE DECLARED VALUE. THEREFORE, NO PROCEDURES WITH ARRAY OR FILE PARAMETERS MAY BE INVOKED WITH A RUN STATEMENT. A PROCEDURE WITH A POINTER PARAMETER, WHETHER OR NOT IT IS DECLARED VALUE, MAY NOT BE INVOKED WITH A RUN STATEMENT.

EXAMPLE:

```
   BEGIN
        TASK TI, TZ;
        INTEGER N;
        PROCEDURE P1 (I);
             INTEGER I; EXTERNAL;
        PROCEDURE P2 (J); VALUE J;
             INTEGER J; EXTERNAL;
(1)     RUN P1(N)  [T1];
(2)     RUN P2(N)  [T2];
   END.
```

STATEMENT (1) WOULD NOT BE ALLOWED, WHEREAS, STATEMENT (2) WOULD BE.

APL-700

D0720 APL-700 - APL CONSOLE COMMANDS - 11-14-74

SEVERAL APL CONSOLE COMMANDS HAVE BEEN IMPLEMENTED. THESE COMMANDS HAVE THE FOLLOWING FORM:

<STACK NO.>SM:)COMMAND

"<STACK NO.>" IS THE JOB NUMBER OF "SYSTEM/APL". "COMMAND" IS ONE OF THE COMMANDS DESCRIBED BELOW.

COMMAND	PURPOSE
UNIT N	N IS AN INTEGER. THIS COMMAND SPECIFIES A CONSOLE UNIT TO WHICH DISPLAYS ARE TO BE DIRECTED. N SHOULD BE THE UNIT NUMBER OF A CONSOLE. IF THIS COMMAND IS NOT USED OR N IS ZERO THEN THE LOWEST NUMBERED UNIT THAT IS A CONSOLE AND IS AVAILABLE WILL BE USED FOR DISPLAYS. IF THE UNIT SPECIFIED IS NOT A CONSOLE AN ERROR WILL BE DISPLAYED AND THE LOWEST NUMBERED CONSOLE UNIT WHICH IS AVAILABLE WILL BE USED.
USERS	DISPLAYS ON THE CONSOLE A LIST OF APL USERS THAT ARE SIGNED ON. THE CONSOLE ON WHICH THE DISPLAY APPEARS IS THE ONE SPECIFIED BY THE "UNIT" COMMAND. THE DISPLAY HAS THE FOLLOWING FORM: LSN USER LSN USER UP TO 20 LINES OF USERS WILL BE DISPLAYED (UP TO 40 USERS). IF MORE THAN 40 USERS ARE CONNECTED THEN "<STACK NO.>SM)NEXT" WILL ALSO BE DISPLAYED IN THE HOME POSITION. IF AN "*" APPEARS BETWEEN THE "LSN" AND "USER" THEN THE USER IS A

COMMAND	PURPOSE
	PRIVILEGED USER.
NEXT	MAY BE USED ONLY WHEN MORE THAN 40 USERS ARE ON AND THE "USERS" COMMAND HAS DISPLAYED THE FIRST SET OF 40 USERS. THIS COMMAND CAUSES THE NEXT SET OF 40 USERS TO BE DISPLAYED. IF MORE THAN 80 USERS ARE ON APL, THEN ENTRY OF "NEXT" WILL GET THE NEXT SET OF 40 ETC.
ADM PARAM	THIS FUNCTIONS IN A SIMILAR MANNER TO THE "ADM" PROVIDED BY THE CONTROLLER. PARAM MAY BE ANY OF THE FOLLOWING: - STOPS "ADM". SI SUSPENDS "ADM". OK RESUMES "ADM" AFTER A SUSPENSION. DISP DISPLAY PARAMETER EMPTY CAUSES THE CURRENT "ADM" PARAMETERS TO BE DISPLAYED. FOR EXAMPLE: DISP IS U D U IS UNIT N OR EMPTY. N IS THE CONSOLE UNIT NUMBER ON WHICH THE "ADM" IS TO BE DISPLAYED. THIS UNIT NEED NOT BE THE SAME AS THE ONE SPECIFIED BY THE "UNIT" COMMAND. IF A UNIT IS NOT SPECIFIED THE ONE SPECIFIED BY THE "UNIT" COMMAND WILL BE USED. D IS (USERS) DELAY K OR EMPTY. THIS CAUSES AN AUTOMATIC DISPLAY OF USERS AT INTERVALS OF K SECONDS. IF MORE THAN 40 USERS ARE ON THEN SUCCESSIVE DISPLAYS WILL SHOW GROUPS OF 40 USERS UNTIL ALL USERS ARE DISPLAYED. NOTE: WHEN THE ADM IS BEING USED THE "USERS" COMMAND WILL NOT BE ACCEPTED.

COMMAND	PURPOSE
WA	DISPLAY ON THE CONSOLE SPECIFIED BY THE "UNIT" COMMAND THE VERSION AND DATE OF COMPILATION OF THE APL BEING RUN.

BACKUP

D0553 BACKUP - PB SECOND REEL - 11-18-73

THIS PATCH CORRECTS BACKUP TO ALLOW THE SECOND (OR GREATER) REEL OF A MULTI-REEL PBT TO BE PRINTED OR PUNCHED WITHOUT HAVING TO READ THROUGH THE FIRST REEL. BECAUSE THERE IS INFORMATION ON THE FIRST REEL WHICH IS NOT PRESENT ON THE OTHERS, THIS DATA WILL HAVE TO BE INPUT BY THE USER. ADDITIONAL INPUT PARAMETERS HAVE BEEN ADDED TO ALLOW THIS INFORMATION TO BE SPECIFIED. THE SYNTAX IS AS FOLLOWS:

?PB MT <normal specifications> REEL = <reel no.> (KIND = {PRINTER / PUNCH}, MODE = {E / B})

THE KIND SPECIFICATION TELLS WHETHER THE PBT IS A PRINTER OR PUNCH BACKUP AND THE MODE SPECIFIES WHETHER IT IS EBCDIC OR BCL.

D0573 BACKUP - PRINTOUT IDENTIFICATION - 12-06-73

TO AID IN IDENTIFYING OUTPUT, BACKUP PRINTS OUT EITHER THE BACKUP FILE NAME (FOR DISK FILES) OR THE TAPE UNIT NUMBER IN BLOCK CHARACTERS PRECEDING THE PRINTOUT. BY SPECIFYING ID="<UP TO 60 CHAR>" AS PART OF THE INPUT STRING, THE ID=STRING WILL BE PRINTED INSTEAD OF THE FILE NAME OR UNIT NUMBER. TO INCLUDE THIS FEATURE, RECOMPILE SYSTEM/BACKUP WITH A $SET IDOPTION CARD IN THE SYMBOLIC.

D0630 BACKUP - TAPE REPOSITIONING - 01-09-73

A NEW FEATURE HAS BEEN ADDED TO BACKUP TO HANDLE CASES OF PRINTER JAMS OR FAILURES DURING LONG PRINTER RUNS, IN WHICH SEVERAL LINES OR PAGES OF OUTPUT FROM A PRINTER BACKUP TAPE HAVE BEEN LOST. THE

OPERATOR MAY INFORM BACKUP TO REPOSITION THE TAPE BY ENTERING

<MIX NO.> HI

AT THE CONSOLE. BACKUP WILL RESPOND WITH

ACCEPT: ENTER SKIP OR UNIT NO.

THE SYNTAX FOR THE REPLY IS AS FOLLOWS:

<mix no.> AX [LP / CP <unit no.>] [+ / -] <skip count> [<channel no.>]

IF THE LP OR CP FOLLOWED BY A UNIT NUMBER IS PRESENT, BACKUP WILL BEGIN USING THIS UNIT INSTEAD OF THE ONE IT WAS PREVIOUSLY USING.

THE FIRST NUMBER WHICH IS SPECIFIED IS A SKIP COUNT. IT MAY BE EITHER POSITIVE OR NEGATIVE, INDICATING SKIPPING FORWARD OR BACKWARD, RESPECTIVELY.

IF NOTHING FOLLOWS THE SKIP COUNT, THEN THE COUNT IS ASSUMED TO REFER TO A NUMBER OF LINES, E.G.,

<MIX NO.> AX -3

MEANS TO BACK UP THREE LINES AND RESUME PRINTING.

IF THE SKIP COUNT IS FOLLOWED BY A NUMBER BETWEEN ONE AND 11, THE NUMBER IS INTERPRETED AS A CHANNEL NUMBER, AND THE SKIP COUNT REFERS TO SKIPS TO THAT PARTICULAR CHANNEL. FOR EXAMPLE:

<MIX NO.> AX -4 3

MEANS TO BACK UP FOUR SKIPS TO CHANNEL THREE.

<MIX NO.> AX 1 1

MEANS TO MOVE FORWARD TO THE NEXT SKIP TO TOP OF PAGE (CHANNEL ONE).

NOTE THAT A SKIP COUNT MUST ALWAYS BE PRESENT, EVEN IF ANOTHER UNIT NUMBER IS SPECIFIED.

D0631 BACKUP - BACKUP FILES ON PACK - 01-09-73

BACKUP MAY BE USED TO PRINT BACKUP FILES WHICH ARE ON PACKS. TO DO THIS, THE INPUT PARAMETERS TO BACKUP SHOULD BE THE SAME AS IF THE FILE WERE ON DISK. HOWEVER, THE PB STATEMENT MUST BE FOLLOWED BY THE FOLLOWING LABEL EQUATION STATEMENTS:

```
FILE FILEIN(KIND=PACK);
FILE F(KIND=PACK);
```

IF THE PACK IS NAMED, THEN PACKNAME=<NAME> SHOULD BE INCLUDED IN THE LABEL EQUATIONS.

D0674 BACKUP - UP-TAPE FILES - 01-27-74

UP-TAPE FILES ON MULTI-FILE REELS MAY NOW BE PRINTED WITHOUT HAVING TO PRINT ALL OF THE PRECEDING FILES. THE NUMBER OF THE FIRST FILE TO PRINT IS SPECIFIED TO BACKUP BY INCLUDING FILE <N> IN THE INPUT STRING. FOR EXAMPLE,

```
<I>PB MT83 FILE 2
```

WOULD CAUSE BACKUP TO BEGIN PRINTING AT THE SECOND FILE ON THE TAPE ON UNIT 83.

D0678 BACKUP - B5700 BACKUP TAPES - 02-03-74

BACKUP MAY NOW BE USED TO PRINT B5700 BACKUP TAPES. WHEN "B5500" OR "B5700" IS SPECIFIED, THE TAPE ON THE SPECIFIED UNIT WILL BE PRINTED, THE OUTPUT GOING TO A NORMAL (I.E., NON-DIRECT) PRINTER FILE CALLED BFILE. ANY OTHER OPTIONS IN THE INPUT STRING (RANGE, LP NO., ETC) WILL BE IGNORED.

EXAMPLE:

```
<I>PB MT 81 B5500
```

D0715 BACKUP - IGNORE INCL CARDS ON SEQ CHECK - 02-20-74

THIS PATCH ALLOWS THE USER TO SPECIFY TO SYSTEM/BACKUP THAT CARDS WHICH HAVE BEEN INCLUDED IN THE LISTING BY AN $INCLUDE CARD ARE NOT TO BE CONSIDERED WHEN MAKING THE SEQUENCE RANGE CHECK, BUT ARE TO BE PRINTED IF THEY FALL WITHIN THE DESIRED SEQUENCE RANGE. THIS FEATURE IS FOR USE ONLY ON LISTINGS GENERATED BY THE ALGOL COMPILER. IT WOULD BE MOST USEFUL IN CASES WHERE INCLUDED CARDS CAUSE THE SYMBOLIC TO BE OUT OF SEQUENCE.

THIS FEATURE IS INVOKED BY PLACING A COLON BETWEEN THE SEQUENCE RANGE LIMITS, FOR EXAMPLE:

```
<I>PB MT 83 KEY ALGOL RANGE 12340000 : 23450000 ;
```

IF A "-" OR BLANKS ARE FOUND BETWEEN THE NUMBERS, PROCESSING IS AS BEFORE. TO SHOW HOW THIS FEATURE COULD BE USEFUL, CONSIDER THE FOLLOWING LISTING:

```
%CARD 1                              00001000
%CARD 2                              10000000
$INCLUDE X 90000000-91000000         10001000
%INCL CARD 1                         90000000
%INCL CARD 2                         90500000
%INCL CARD 3                         91000000
%CARD 3                              10002000
%CARD 4                              10003000
%CARD 5                              20000000
```

IF IT WERE DESIRED TO PRINT THE CARDS IN THE SEQUENCE RANGE 1000000 - 19999999, THE FOLLOWING STATEMENT COULD BE USED:

```
<I>PB <DEVICE SPECIFIER> KEY ALGOL RANGE 10000000 : 19999999 ;
```

NOTE THAT IF BACKUP WERE RUN WITHOUT USING THE NEW FEATURE, IT WOULD STOP PRINTING AS SOON AS IT HIT THE CARD AT SEQ 90000000.

BASIC

D0584 BASIC - DOLLAR OPTION "PAGE" - 11-18-73

THE DOLLAR CARD OPTION "PAGE" HAS NOW BEEN PROPERLY IMPLEMENTED IN B6700 BASIC. IF THE DOLLAR OPTION "LIST" IS SET WHEN COMPILING A BASIC PROGRAM, THEN EVERY OCCURRENCE OF THE DOLLAR OPTION "PAGE" WILL CAUSE THE COMPILER TO SKIP TO THE TOP OF THE NEXT PAGE OF THE OUTPUT LISTING.

D0585 BASIC - MATRIX I-O - 11-18-73

THE BASIC COMPILER WILL NOW GIVE THE SYNTAX ERROR "MATRIX IDENTIFIER EXPECTED" IF THE LIST OF A "MAT PRINT", "MAT READ" OR "MAT INPUT" STATEMENT CONTAINS ANY TOKEN OTHER THAN A MATRIX IDENTIFIER OR A LIST DELIMITER.

NOTE: THIS ERROR MESSAGE WAS NOT BEING GENERATED FOR PROGRAMS COMPILED WITH PRE-2.5 BASIC OR 2.5 BASIC WITH THE OPTION "OLDBASIC" SET, AND BAD CODE WAS BEING GENERATED.

D0586 BASIC - STRING VARIABLE NAMES - 11-18-73

THE BASIC COMPILER NOW PROPERLY HANDLES STRING VARIABLE NAMES WHICH CONSIST OF AN ALPHABETIC CHARACTER FOLLOWED BY A DIGIT FOLLOWED BY A DOLLAR SIGN (E.G., A7$).

D0587 BASIC - PRINT STATEMENTS - 11-18-73

IF THE LAST PRINT STATEMENT IN A BASIC PROGRAM IS TERMINATED WITH A COLON, THIS LAST LINE OF OUTPUT WILL NOW BE PRINTED AT END-OF-JOB.

D0587 BASIC - PRINT STATEMENTS - 11-18-73

NOTE: THIS ONLY HAS SIGNIFICANCE FOR PROGRAMS COMPILED WITH PRE-2.5 BASIC OR FOR PROGRAMS COMPILED WITH 2.5 BASIC WITH THE OPTION "OLDBASIC" SET.

D0588 BASIC - RELATIONAL OPERATORS - 11-18-73

THE BASIC COMPILER NOW PERMITS THE PROGRAMMER TO USE AN IDENTIFIER NAME BEGINNING WITH THE LETTER "Q" FOLLOWING ANY RELATIONAL OPERATOR, INCLUDING "LE", "GE", AND "NE".

D0589 BASIC - SINGLE LINE "DEF" FUNCTIONS - 11-18-73

A PROGRAM COMPILING UNDER BASIC WILL NO LONGER BE TERMINATED WITH AN INVALID INDEX FAULT IF IT CONTAINS A PARAMETERLESS "DEF" FUNCTION, WHERE THAT FUNCTION DEFINITION CAUSES A NEW STACK LOCATION TO BE ALLOCATED (I.E., THE FUNCTION IS THE FIRST STATEMENT IN THE PROGRAM TO REFERENCE A CERTAIN VARIABLE).

D0596 BASIC - SEGMENTATION OF CODE FILES - 12-28-73

PLEASE REFER TO THE NEW BASIC LANGUAGE MANUAL FOR AN EXPLANATION OF SEGMENTATION OF CODE FILES.

D0683 BASIC - FILES WITH ODD LENGTH RECORDS - 01-27-74

CURRENTLY CANDE CANNOT WRITE OR UPDATE CHARACTER MODE FILE WHICH HAS A MAXRECSIZE THAT IS AN ODD NUMBER. FILES CREATED BY "PRINT" AND "WRITE" STATEMENTS IN A BASIC PROGRAM ARE CREATED AS CHARACTER MODE FILES WITH A MAXRECSIZE OF 81, AND THUS CANNOT BE MODIFIED BY CANDE. THEREFORE, IF THE USER CREATES FILES WITH BASIC PRINT AND WRITE STATEMENTS WHICH HE PLANS TO MANIPULATE THROUGH CANDE, HE MUST CREATE THE FILES WITH AN EVEN NUMBER OF CHARACTERS FOR THE

RECORD SIZE. THIS CAN BE DONE IN THE BASIC PROGRAM BY USING THE "MARGIN" STATEMENT.

EXAMPLE:

```
MARGIN #1, 82
```

THIS PROBLEM WILL BE CORRECTED IN A SUBSEQUENT RELEASE OF CANDE.

COBOL

D0660 COBOL - CONDITION NAMES - 11-18-73

B6700 COBOL NOW ALLOWS CONDITION-NAMES TO BE DECLARED ASSOCIATED WITH GROUP DATA ITEMS. SUCH CONDITION-NAMES, OF COURSE, ARE ALPHANUMERIC IN CLASS AND MAY NOT USE NUMERIC LITERALS IN THEIR VALUE CLAUSES.

D0662 COBOL - FILLER ITEMS - 12-28-73

THE 2.5 COMPILER DID NOT ALLOW NUMERIC FILLER ITEMS TO BE LARGER THAN 23 DECIMAL PLACES. THIS WAS INCORRECT. THESE ITEMS ARE NOW ALLOWED UNLESS THEY HAVE VALUE CLAUSES OR CONDITION-NAMES ASSOCIATED WITH THEM.

D0663 COBOL - "COMP-1" ARRAYS - 01-06-74

THE COMPILER WAS SOMEWHAT LAX ON ALLOWING CONFLICTING DATA DESCRIPTION CLAUSES IN THE DESCRITPION OF "COMP-1" STACK ARRAYS. THE COMPILER NOW PROHIBITS "OWN", "GLOBAL", "RECORD AREA", "SEGMENT", "LOWER-BOUND", AND "OCCURS" CLAUSES WITH COMP-1 STACK ARRAYS. THESE ARRAYS MAY BE DECLARED ONLY IN THE WORKING-STORAGE SECTION.

D0664 COBOL - "MYJOB" CONTROL-POINT - 01-20-74

"MYJOB", A RESERVED CONTROL-POINT IDENTIFIER (TASK VARIABLE), IS AVAILABLE IN COBOL. "MYJOB" REFERS TO THE TASK VARIABLE ATTACHED TO THE JOB (OF WHICH THE TASK IS A PART) AND MAKES THE

D0664 COBOL - "MYJOB" CONTROL-POINT - 01-20-74

CHARACTERISTICS OF THE JOB AVAILABLE TO THE TASK IN THE SAME WAY AS "MYSELF" MAKES AVAILABLE THE CHARACTERISTICS OF THE TASK.

D0684 COBOL - DECLARATIVE PROCEDURES - 11-18-73

THIS PATCH ALLOWS "GO" STATEMENTS IN A DECLARATIVE ERROR OR LABEL PROCEDURE TO REFERENCE THE LABELS OF ANOTHER DECLARATIVE ERROR OR LABEL PROCEDURE. THE "GO" STATEMENT MAY APPEAR ONLY IN AN ERROR OR LABEL "USE" PROCEDURE, AND MAY REFERENCE ONLY AN ERROR OR LABEL "USE" PROCEDURE. WHEN EXECUTION REACHES THE END OF ANY ERROR OR LABEL "USE" PROCEDURE, TRANSFER OF CONTROL OCCURS BACK TO THE STATEMENT FOLLOWING THE STATEMENT WHICH CAUSED EXECUTION OF THE "USE" PROCEDURE EVEN WHEN "GO" STATEMENTS HAVE CAUSED TRANSFER OF CONTROL TO OTHER "USE" PROCEDURES.

D0685 COBOL - TWO DIMENSIONAL ARRAYS - 11-18-73

TWO DIMENSIONAL ARRAYS (01 ITEMS HAVING AN OCCURS CLAUSE) ARE NOW ALLOWED IN THE LINKAGE SECTION.

D0686 COBOL - DISPLAY STATEMENT - 12-06-73

IN ADDITION TO LITERALS, FIGURATIVE CONSTANTS, AND DATA ITEMS, IT IS NOW POSSIBLE TO DISPLAY THE VALUES OF SPECIAL REGISTERS AND ATTRIBUTES WHOSE IMPLICIT CLASS IS NUMERIC. THE NUMBER OF DECIMAL PLACES DISPLAYED IS UNDEFINED BUT USUALLY WILL BE SIX.

D0687 COBOL - MOVE STATEMENTS - 01-06-74

PAGE 7-67 OF THE COBOL REFERENCE MANUAL, IN THE SYNTAX FOR OPTION ONE OF THE MOVE STATEMENT, SPECIFIES ONLY LITERALS, FIGURATIVE CONSTANTS, AND DATA IDENTIFIERS AS PERMISSIBLE SOURCE OPERANDS.

D0687 COBOL - MOVE STATEMENTS - 01-06-74

THE MANUAL SHOULD BE CHANGED TO INCLUDE SPECIAL REGISTERS AND ATTRIBUTES AS PERMISSIBLE SOURCE OPERANDS. SPECIAL REGISTERS ARE DEFINED ON PAGE 2-10 OF THE COBOL MANUAL. THE LIST OF SPECIAL REGISTERS SHOULD INCLUDE, IN ADDITION TO THOSE ALREADY SPECIFIED: CHECKPOINT-STATUS, THE FILE, TASK, AND DIRECT I/O ATTRIBUTES, AND THE MNEMONIC VALUE FUNCTIONS. MOST SPECIAL REGISTERS HAVE AN IMPLICIT CLASS OF NUMERIC. POINTER, TASK, AND EVENT TYPED ATTRIBUTES ARE THE EXCEPTION. NUMERIC SPECIAL REGISTERS AND ATTRIBUTES CAN BE USED NEARLY ANYWHERE IN THE SYNTAX OF THE PROCEDURE DIVISION THAT A NUMERIC VALUE IS ACCEPTABLE, SUCH AS SOURCE OPERANDS IN MOVE, ADD, AND SUBTRACT STATEMENTS. MOVING NUMERIC SPECIAL REGISTERS AND ATTRIBUTES TOALPHANUMERIC RECEIVING FIELDS CAN PRODUCE INTERESTING RESULTS BECAUSE OF ELEMENTARY MOVE RULE FOUR GIVEN ON PAGE 7-69. THE LENGTH OF THE SOURCE VALUE IS UNDEFINED. FOR THIS REASON, IT IS BEST TO MOVE NUMERIC SPECIAL REGISTERS AND ATTRIBUTES TO NUMERIC OR NUMERIC-EDITED RECEIVING FIELDS TO ACHIEVE DECIMAL POINT ALIGNMENT AND HIGH-ORDER TRUNCATION. THE COMPILER NOW EMITS A WARNING MESSAGE TO HELP AVOID THE PITFALLS OF USING AN ALPHANUMERIC RECEIVING FIELD.

D0688 COBOL - SEARCH STATEMENT - 10-20-74

THIS PATCH IMPROVES THE EFFICIENCY OF THE SEARCH STATEMENT AND IMPLEMENTS THE "SEARCH ALL" CONSTRUCT. THIS INITIAL IMPLEMENTATION OF SEARCH ALL WILL GENERATE A SERIAL SEARCH OF THE TABLE, STARTING AT THE FIRST ENTRY. NO SYNTACTICAL CHECK IS MADE WITH REGARD TO THE OPERANDS AND COMPARISONS IN THE WHEN CONDITION. THE SYNTAX FOR THE SEARCH ALL STATEMENT IS:

```
SEARCH ALL <identifier>  [; AT END  <imperative-statement>]
                         ; WHEN  <condition>  { <imperative-statement> }
                                              { NEXT SENTENCE          }
```

THE IDENTIFIER MUST NOT BE SUBSCRIPTED OR INDEXED, BUT ITS

D0688 COBOL - SEARCH STATEMENT - 10-20-74

DESCRIPTION MUST CONTAIN AN OCCURS CLAUSE WITH AN INDEXED PHRASE AND A KEY PHRASE.

THE WHEN CONDITION SHOULD BE CONSTRUCTED SO THAT EACH KEY ITEM IS PART OF A RELATION CONDITION. ONLY DATA-NAMES WHICH APPEAR IN THE KEY CLAUSE OF AN <IDENTIFIER> SHOULD BE TESTED IN THE "WHEN" CLAUSE.

THE PURPOSE OF THE WHEN CONDITION IS TO PROVIDE AN ALGORITHM FOR DETERMINING WHEN TO STOP SEARCHING THE TABLE.

D0689 COBOL - OWN INDEX-NAMES - 01-20-74

ON 2.5 COBOL, THE INDEX-NAMES DECLARED IN ASSOCIATION WITH THE DATA ITEMS OF OWN OR GLOBAL TABLES (ARRAYS) WERE GIVEN AN INCORRECT ADDRESS BY THE COMPILER AND INSUFFICIENT INFORMATION WAS GENERATED FOR THE BINDER TO KNOW OF THE OWN INDEX-NAMES, CAUSING UNDEFINED RESULTS. THE INDEX-NAMES ASSOCIATED WITH GLOBAL ITEMS ARE TREATED AS OWN; THEY DO NOT LOSE THEIR VALUE UPON EXIT FROM THE SUBROUTINE IN WHICH THEY ARE DECLARED.

D0690 COBOL - CONDITIONAL STATEMENTS - 01-20-74

B6700 COBOL, AS WELL AS OTHER BURROUGHS COBOL LANGUAGES, HAS ALWAYS TREATED THE AT END, INVALID KEY, ON SIZE ERROR, AT END-OF-PAGE, AND ON EXCEPTION CLAUSES AS IMPLICIT CONDITIONS TO WHICH AN ELSE WOULD BE MATCHED IF ONE APPEARED. THIS CONFLICTS WITH THE REQUIREMENTS OF THE USASI 1968 COBOL STANDARD, AS WELL AS THE COBOL LANGUAGES OF OTHER MANUFACTURERS. IN AN EFFORT TO IMPROVE COMPATIBILITY IN THIS AREA, THE SYSTEM OPTIONS USASI OR S360 MAY BE USED TO CAUSE ELSE"S TO BE MATCHED ONLY TO IF STATEMENTS.

D0691 COBOL - SERIALNO ATTRIBUTE - 01-20-74

BECAUSE THE SERIALNO FILE ATTRIBUTE HAS BEEN CHANGED WITHIN THE SYSTEM FROM AN ARITHMETIC INTEGER VALUE TO A SIX-CHARACTER ALPHANUMERIC OPERAND, IT IS NO LONGER AVAILABLE, WITHIN THE COBOL LANGUAGE, AS A NUMERIC OPERAND, OR, AS AN ALPHANUMERIC OPERAND IN A CONDITIONAL EXPRESSION. AS A READ-ONLY ATTRIBUTE, ITS USE IS RESTRICTED TO BEING THE SENDING ITEM IN A MOVE STATEMENT. THERE IT IS CONSIDERED TO BE AN ALPHANUMERIC OPERAND WITH A LENGTH OF SIX; IT MAY NOT BE MOVED TO A NUMERIC OR NUMERIC EDITED RECEIVING FIELD.

D0692 COBOL - ARITHMETIC EXPRESSIONS - 01-27-74

UNDER MANY CIRCUMSTANCES, THE COMPILER ACCEPTED NON-NUMERIC LITERALS AS OPERANDS IN ARITHMETIC EXPRESSIONS, TREATING THEM AS NUMERIC LITERALS, STRIPPING ZONES, ETC. IT MIGHT BE PROFITABLE TO SPECIFY EXACTLY WHAT TYPES OF OPERANDS MAY APPEAR IN ARITHMETIC FORMULAS. PAGE 7-11 OF THE COBOL MANUAL IS SOMEWHAT GENERAL ON THIS POINT. BELOW ARE LISTED THE TYPES OF OPERANDS.

1. NUMERIC DATA ITEMS
2. NUMERIC LITERALS
3. INTRINSIC FUNCTIONS (INCLUDING ALGEBRAIC TYPED INSTALLATION INTRINSICS)
4. THE FIGURATIVE CONSTANT ZERO, ZEROES, ZEROS
5. SPECIAL REGISTERS AND ATTRIBUTES WHOSE CLASS IS IMPLICITLY NUMERIC; INCLUDED IN THIS CATEGORY ARE:
   A. THE TIME AND COMPILETIME FUNCTIONS
   B. TODAYS-DATE
   C. TALLY
   D. CHECKPOINT-STATUS

E. LINAGE-COUNTER

F. THE ATTRIBUTE MNEMONIC VALUE FUNCTIONS

G. ANY FILE, TASK, OR DIRECT-IO AREA ATTRIBUTE WHOSE IMPLICIT CLASS IS NUMERIC

H. THE VARIOUS DATA MANAGEMENT STATUS FUNCTIONS

D0693 COBOL - WRITE BLOCK - 02-03-74

THE COMPILER NOW EMITS A SYNTAX ERROR WHENEVER THE WRITE BLOCK CONSTRUCT IS USED. THE USE OF THIS CONSTRUCT HAS ALWAYS PRODUCED UNDEFINED RESULTS. THERE ARE NO KNOWN REASONS FOR KEEPING THIS CONSTRUCT IN THE COBOL LANGUAGE.

D0698 COBOL - INSTALLATION INTRINSICS - 12-28-73

UNTYPED INSTALLATION INTRINSICS MAY NOW BE CALLED VIA OPTION TWO OF THE CALL STATEMENT. THE REVISED SYNTAX FOR OPTION TWO OF THE CALL STATEMENT IS:

CALL { section-name / installation-intrinsic } [USING actual-parameter-list]

ACTUALLY EITHER TYPED OR UNTYPED INSTALLATION INTRINSICS MAY BE CALLED. HOWEVER, THE VALUE RETURNED BY TYPED INTRINSICS WILL BE DELETED. TYPED INSTALLATION INTRINSICS CONTINUE, OF COURSE, TO BE AVAILABLE IN ARITHMETIC OR CONDITIONAL EXPRESSIONS. BOOLEAN INTRINSICS ARE TREATED AS BOOLEAN WHEN USED IN A CONDITIONAL EXPRESSION, AND AS INTEGER FUNCTIONS WHEN USED IN ARITHMETIC EXPRESSIONS (A WARNING MESSAGE IS NOW ISSUED FOR THIS CASE).

SEVERAL RELATED SUBJECTS SHOULD BE CLARIFIED. THE SYNTAX OF THE RUN STATEMENT IS INCORRECT AND SHOULD BE CHANGED TO:

RUN <CONTROL-POINT-IDENTIFIER> WITH <SECTION-NAME>

[USING <ARITHMETIC-EXPRESSION-1>

[, <ARITHMETIC-EXPRESSION-2>]...]

THE RUN STATEMENT INITIATES INDEPENDENT TASKS, TASKS WHICH MUST NOT BE DEPENDENT UPON THE EXISTENCE OF THE INITIATING TASK. CONSEQUENTLY, ALL PARAMETERS MUST BE PASSED BY VALUE, I.E., RECEIVED BY CONTENT. IN COBOL, ONLY ARITHMETIC VALUES MAY BE PASSED OR RECEIVED, SINCE ONLY FORMAL PARAMETERS HAVING ARITHMETIC PROPERTIES CAN BE DESCRIBED IN THE COBOL SYNTAX. THE FORMAL PARAMETERS, TO WHICH THE VALUES OF THE ARITHMETIC EXPRESSIONS ARE PASSED, MUST BE DESCRIBED AS SINGLE OR EXTENDED PRECISION 77-LEVEL COMP, COMP-1, COMP-4, OR COMP-5 ITEMS AND SHOULD HAVE A "RECEIVED BY CONTENT" CLAUSE, EVEN THOUGH "RECEIVED BY CONTENT" IS THE DEFAULT. THE COMPILER MAKES ANY NECESSARY ADJUSTMENTS, IF NECESSARY, TO TRUNCATE EXTENDED PRECISION VALUES TO SINGLE PRECISION, OR EXTENDED SINGLE PRECISION VALUES TO INSURE THAT THE VALUE PASSED HAS THE SAME PRECISION AS THE CORRESPONDING FORMAL PARAMATER. ALL VALUES ARE PASSED WITH A SCALE OF ZERO, REGARDLESS OF THE SCALE OF THE CORRESPONDING FORMAL PARAMETER, AND MAY BE PASSED AS NORMALIZED VALUES.

THE SYNTAX OF THE PROCESS STATEMENT AND OPTION ONE OF THE CALL STATEMENT SHOULD BE CHANGED TO:

CALL <CONTROL-POINT-IDENTIFIER> WITH <SECTION-NAME>

[USING <ACTUAL-PARAMETER-LIST>]

PROCESS <CONTROL-POINT-IDENTIFIER> WITH <SECTION-NAME>

[USING <ACTUAL-PARAMETER-LIST>]

THE <ACTUAL-PARAMETER-LIST> MUST CONSIST OF A SERIES OF DATA-ITEMS, CONTROL-ITEMS, AND EXPRESSIONS, OPTIONALLY SEPARATED BY COMMAS. IN ADDITION TO PASSING ARITHMETIC VALUES, CERTAIN KINDS OF VARIABLES MAY BE PASSED (RECEIVED) BY REFERENCE. AS A GENERAL RULE, THE KIND OF ACTUAL PARAMETER MUST NOT CONFLICT WITH THE CORRESPONDING FORMAL PARAMETER. THE FOLLOWING LISTS ALL PERMISSIBLE KINDS OF FORMAL

PARAMETERS THAT CAN BE DECLARED IN COBOL, THE CORRESPONDING DECLARATION IN ALGOL, AND THE PERMISSIBLE KINDS OF ACTUAL PARAMETERS THAT MAY BE PASSED.

	COBOL FORMAL PARAMETER	ALGOL FORMAL PARAMETER	PERMISSIBLE ACTUAL PARAMETER
1.	77-LEVEL SINGLE PRECISION COMP, COMP-1, OR COMP-4 ITEM (RECEIVED BY CONTENT)	REAL OR INTEGER	ARITHMETIC-EXPRESSION
2.	77-LEVEL EXTENDED PRECISION COMP, COMP-1, OR COMP-5 ITEM (RECEIVED BY CONTENT)	DOUBLE	ARITHMETIC-EXPRESSION
3.	77-LEVEL SINGLE PRECISION COMP, COMP-1, OR COMP-4 ITEM (RECEIVED BY REFERENCE)	REAL OR INTEGER	77-LEVEL SINGLE PRECISION COMP, COMP-1,OR COMP-4 ITEM
4.	77-LEVEL EXTENDED PRECISION COMP, COMP-1,OR COMP-5 ITEM (RECEIVED BY REFERENCE)	DOUBLE	77-LEVEL EXTENDED PRECISION COMP, COMP-1, OR COMP-5 ITEM
5.	77-LEVEL EVENT OR LOCK ITEM	EVENT	77-LEVEL EVENT OR LOCK ITEM
6.	77-LEVEL CONTROL-POINT ITEM	TASK	77-LEVEL CONTROL-POINT ITEM
7.	77-LEVEL INDEX FILE ITEM	DIRECT SWITCH FILE	77-LEVEL INDEX FILE ITEM
8.	FILE	FILE	FILE
9.	DIRECT FILE	DIRECT FILE	DIRECT FILE
10.	01-LEVEL EVENT	EVENT ARRAY	01-LEVEL EVENT OR

	COBOL FORMAL PARAMETER	ALGOL FORMAL PARAMETER	PERMISSIBLE ACTUAL PARAMETER
	OR LOCK ITEM		LOCK ITEM
11.	01-LEVEL CONTROL-POINT ITEM	TASK ARRAY	01-LEVEL CONTROL-POINT ITEM
12.	01-LEVEL COMPUTATIONAL ITEM	ARRAY, INTEGER OR REAL	*01-LEVEL COMP, COMP-2, DISPLAY, OR DISPLAY-1 ITEM
13.	01-LEVEL COMP-2 ITEM	HEX ARRAY	*01-LEVEL COMP, COMP-2, DISPLAY, OR DISPLAY-1 ITEM
14.	01-LEVEL DISPLAY-1 ITEM	BCL ARRAY	*01-LEVEL COMP, COMP-2, DISPLAY, OR DISPLAY-1 ITEM
15.	01-LEVEL DISPLAY ITEM	EBCDIC ARRAY	*01-LEVEL COMP, COMP-2, DISPLAY, OR DISPLAY-1 ITEM

SPECIAL RULES:

1. ASCII ARRAYS ARE CONSIDERED THE SAME AS EBCDIC ARRAYS FOR PARAMETER PASSING PURPOSES.
2. DIRECT ARRAYS MAY BE PASSED TO NON-DIRECT ARRAYS, BUT NOT VICE VERSA.
3. *THE COBOL COMPILER WILL CHANGE THE CHARACTER TYPE AND LENGTH OF AN ARRAY DESCRIPTOR, IF NECESSARY, TO MATCH THE CHARACTER TYPE OF THE FORMAL PARAMETER, FOR ONE-DIMENSIONAL ARRAYS ONLY. THE CHARACTER TYPE OF A TWO-DIMENSIONAL ARRAY MUST MATCH THAT OF THE FORMAL PARAMETER.
4. ANY ITEM THAT MAY BE PASSED BY REFERENCE MAY BE DECLARED GLOBAL.
5. "STACK" ARRAYS (COMP-1 01"S) MAY NOT BE PASSED AS PARAMETERS.
6. CONDITIONAL EXPRESSIONS MAY BE PASSED TO INSTALLATION INTRINSICS HAVING FORMAL VALUE BOOLEAN PARAMETERS; ANY DATA ITEM MAY BE PASSED TO INSTALLATION INTRINSICS HAVING FORMAL VALUE POINTER

PARAMETERS.

7. A SUBSCRIPTED 01-LEVEL ITEM WHICH HAS AN OCCURS CLAUSE MAY BE PASSED TO A ONE-DIMENSIONAL ARRAY.

8. THE COBOL COMPILER RESERVES THE RIGHT TO PREVENT ELEMENTARY 01-LEVEL NUMERIC ITEMS OF ANY USAGE FROM BEING CONSIDERED "ARRAYS".

D0700 COBOL - TIME DOLLAR OPTION - 11-28-73

IF THE TIME DOLLAR OPTION IS SET, THE COMPILER WILL LIST THE NORMAL HEADING AND FOOTING OF THE COMPILE LISTING EVEN THOUGH LIST WAS NEVER SET.

D0701 COBOL - WRITE STMT FOR REMOTE FILES - 12-28-73

THE ALLOWABLE SYNTAX FOR WRITE STATEMENTS TO FILES ASSIGNED TO REMOTE DEVICES HAS BEEN EXPANDED. ALSO, THE SYNTACTICAL DEFINITION OF THE WRITE STATEMENT IN THE DATA COMMUNICATIONS SECTION OF THE COBOL REFERENCE MANUAL ON PAGE 10-6 WAS INCORRECT. LINE SPACING CONTROL HAS NEVER BEEN AVAILABLE UNTIL NOW. THE NEW SYNTAX FOR FILES ASSIGNED TO REMOTE IS:

OPTION ONE:

WRITE <RECORD-NAME> [FROM <IDENTIFIER>]

[;INVALID KEY <STATEMENT> [ELSE <STATEMENT>]]

OPTION TWO:

WRITE <RECORD-NAME> [FROM <IDENTIFIER>]

$$\left\{\begin{matrix}\underline{\text{BEFORE}}\\ \underline{\text{AFTER}}\end{matrix}\right\} \text{ ADVANCING TO } \left\{\begin{matrix}\underline{\text{PAGE}}\\ \underline{\text{CHANNEL}}\ \text{<integer>}\\ \text{<numeric-operand>}\ \text{LINES}\\ \text{<mnemonic-channel-name>}\end{matrix}\right\}$$

PREVIOUSLY, ONLY OPTION ONE, WITH THE "INVLAID KEY" CLAUSE WAS ALLOWED. THE SELECT CLAUSE SYNTAX ON PAGE 10-2 SHOULD ALSO BE CORRECTED. "DIRECTED" ACCESS WAS NEVER IMPLEMENTED IN THE DATACOM I/O SYSTEM, AND THE COBOL SYNTAX FOR IT WAS DELETED ON THE MARK 2.5 RELEASE. SPECIFICATION OF SEQUENTIAL ACCESS IS, ESSENTIALLY, NOW A NO-OP. THE CORRECT SYNTAX OF THE SELECT CLAUSE FOR REMOTE FILES IS:

SELECT <FILE-NAME> ASSIGN TO REMOTE

[,ACCESS MODE IS SEQUENTIAL]

[,ACTUAL KEY IS <DATA-NAME>]

FILES DECLARING ACTUAL KEYS HAVE THE ACTUAL KEY SET TO THE RELATIVE STATION NUMBER WHEN A READ IS DONE. WRITE STATEMENTS FOR THIS FILE ARE THEN "KEYED" WRITES, I.E., THE VALUE OF THE ACTUAL KEY IS PASSED TO LOGICAL-I/O, DIRECTING THE RECORD TO THE SPECIFIED STATION. THIS ACTION WILL CONTINUE TO TAKE PLACE, EVEN THOUGH AN OPTION TWO WRITE STATEMENT IS USED. THE COMPILER WILL GENERATE AN IMPLICIT CALL ON "ATTRIBUTE HANDLER" SETTING FILE(LASTSTATION) TO THE VALUE OF THE ACTUAL KEY JUST BEFORE CALLING FOR THE I/O.

D0702 COBOL - USER LABEL RECORDS - 01-06-74

THIS CHANGE REMOVES THE RESTRICTION THAT LABEL RECORDS MUST ALL APPEAR PRIOR TO THE FIRST RECORD DESCRIPTION. IT ALSO CORRECTS A PROBLEM IN WHICH THE DECLARATION OF MORE THAN NINE LABEL RECORDS CAUSED THE COMPILER TO TERMINATE WITH AN INVALID INDEX.

D0703 COBOL - CHECKPOINT STATEMENT - 01-20-74

PART OF THE NEW CHECKPOINT-RESTART FACILITY IS THE CHECKPOINT STATEMENT IN COBOL. THE SYNTAX IS:

CHECKPOINT [TO {DISK / DISKPACK}] [WITH {LOCK / PURGE}] [;ON EXCEPTION <statement> [ELSE <statement>]]

THE SPECIAL-REGISTER, CHECKPOINT-STATUS, WILL CONTAIN THE RESULT WORD OF THE CHECKPOINT.

D0704 COBOL - SEGMENT-LIMIT CLAUSE - 01-20-74

THE COMPILER NOW LIMITS THE VALUE TO WHICH SEGMENT-LIMIT CAN BE SET, TO 4095 WORDS.

D0705 COBOL - LINAGE-COUNTER - 01-20-74

THE LINAGE CLAUSE NOW WORKS CORRECTLY. THERE ARE SEVERAL CORRECTIONS THAT MUST BE MADE, HOWEVER, TO THE COBOL REFERENCE MANUAL WITH REGARD TO THE LINAGE CLAUSE AND ITS SPECIAL REGISTER, THE LINAGE-COUNTER.

1. ON PAGE 2-11, THE SECOND SENTENCE OF THE "LINAGE-COUNTER" PARAGRAPH SHOULD BE CHANGED TO READ: THE IMPLICIT CLASS OF A LINAGE-COUNTER IS NUMERIC.
2. ON PAGE 6-26, "RULE B" SHOULD BE CHANGED TO: A LINAGE-COUNTER MAY BE USED AS A NUMERIC SOURCE OPERAND BY PROCEDURE DIVISION STATEMENTS. IF MORE THAN ONE FILE HAS A LINAGE CLAUSE, THEN ALL REFERENCES TO LINAGE-COUNTER MUST BE QUALIFIED BY THE FILE NAME.
3. ON PAGE 6-26 "RULE C" SHOULD BE CHANGED TO: THE LINAGE-

D0705 COBOL - LINAGE-COUNTER - 01-20-74

COUNTER MAY NOT BE USED AS A RECEIVING OPERAND BY PROCEDURE DIVISION STATEMENTS.

4. ON PAGE 6-27, DELETE THE NEXT TO LAST SENTENCE.

THE PRESENCE OF A LINAGE CLAUSE CAUSES THE PAGESIZE ATTRIBUTE TO BE SET TO THE VALUE OF THE LITERAL OR DATA-NAME AT THE TIME THE FILE IS OPENED. THE LINAGE-COUNTER OF A FILE IS SYNONOMOUS WITH THE LINENUM ATTRIBUTE.

THE "AT END-OF-PAGE" CLAUSE MAY NOW BE USED WITH WRITE STATEMENTS FOR ANY FILE ASSIGNED TO A PRINTER, REGARDLESS OF WHETHER OR NOT A LINAGE CLAUSE IS DECLARED WITH THE FILE. THE END-OF-PAGE CONDITON WILL OCCUR AS THE LAST LINE IS WRITTEN, IF LINAGE IS SPECIFIED OR THE ATTRIBUTE PAGESIZE IS SET TO SOME VALUE OTHER THAN ZERO.

D0708 COBOL - INTRINSIC FUNCTIONS - 01-20-74

THE ARITHMETIC INTRINSIC FUNCTIONS "ONES", "FIRSTONE", "MAX", AND "MIN" ARE NOW AVAILABLE IN COBOL. THEY HAVE BEEN RESERVED WORDS FOR SOME TIME, BUT NOT IMPLEMENTED AS INTRINSIC FUNCTIONS. ONES AND FIRSTONE REQUIRE ONE PARAMETER. MAX AND MIN REQUIRE AT LEAST ONE PARAMETER. INTRINSIC FUNCTIONS ARE AVAILABLE AT ANY POINT IN THE SYNTAX THAT AN ARITHMETIC EXPRESSION IS PERMITTED.

1. ONES RETURNS THE NUMBER OF NON-ZERO BITS IN THE PARAMETER VALUE. THE PARAMETER MAY BE EITHER A SINGLE OR DOUBLE PRECISION EXPRESSION.
2. FIRSTONE RETURNS THE BIT NUMBER, PLUS ONE, OF THE LEFTMOST NON-ZERO BIT IN THE PARAMETER VALUE. IF THE PARAMETER VALUE IS DOUBLE PRECISION ONLY THE FIRST (HIGHER ORDER) WORD IS USED. IF THE PARAMETER VALUE IS ZERO, FIRSTONE RETURNS ZERO.
3. MAX RETURNS THE MAXIMUM OF THE PARAMETER VALUES.
4. MIN RETURNS THE MINIMUM OF THE PARAMETER VALUES.

D0709 COBOL - FILE DESCRIPTIONS - 02-03-74

SEVERAL CHANGES HAVE BEEN MADE REGARDING THE WAY THE COMPILER INTERPRETS FILE DESCRIPTIONS. THESE CHANGES MAKE B6700 COBOL MORE COMPATIBLE WITH THE 1968 USASI COBOL STANDARD AND WITH OTHER IMPLEMENTATIONS OF COBOL. THE MOST SIGNIFICANT CHANGE ELIMINATES THE "RECORD CONTAINS" CLAUSE AS A FACTOR IN THE DETERMINATION OF THE "MAXRECSIZE" AND "FILETYPE" ATTRIBUTES.

THE REQUIREMENT THAT THE FIRST DATA RECORD MUST BE AS LARGE AS ANY OTHER DATA RECORD HAS BEEN ELIMINATED. THE USAGE OF THE FIRST DATA RECORD CONTINUES, OF COURSE, TO DETERMINE THE INTERNAL CHARACTER SIZE (INTMODE ATTRIBUTE) OF THE FILE. THE MAXIMUM RECORD SIZE IS NOW OBTAINED FROM THE LENGTH OF THE LARGEST RECORD, IN TERMS OF THE INTERNAL CHARACTER SIZE OF THE FILE. THE PRESENCE OF A "RECORD CONTAINS" CLAUSE IS IGNORED IN THIS CALCULATION. PREVIOUSLY, THE LARGER OF THE TWO VALUES WAS TAKEN AS THE MAXIMUM RECORD SIZE. USUALLY, OF COURSE, THERE WAS NO CONFLICT.

FOR FILES WHOSE FILETYPE IS ZERO (FIXED LENGTH RECORDS), HAVING MULTIPLE RECORD DESCRIPTIONS OF DIFFERING LENGTHS, WRITE STATEMENTS WILL PASS, AS THE "SIZE" PARAMETER TO LOGICAL I/O, THE "LENGTH" (IN TERMS OF INTERNAL CHARACTER SIZE) OF THE RECORD REFERENCED IN THE WRITE STATEMENT INSTEAD OF THE MAXIMUM RECORD SIZE. THIS WILL ONLY LIMIT THE AMOUNT OF DATA TRANSFERRED FROM THE RECORD AREA TO THE BUFFER, NOT THE AMOUNT TRANSFERRED TO THE I/O DEVICE, EXCEPT FOR UNBLOCKED FILES ASSINGED TO CERTAIN DEVICE TYPES LIKE PRINTERS.

THE "BLOCK CONTAINS" CLAUSE OF THE FILE DESCRIPTION AND THE "SIZE" CLAUSE OF THE 01-LEVEL RECORD DESCRIPTIONS ARE NOW THE ONLY BASIS FOR DETERMINING THE "FILETYPE" ATTRIBUTE. THE PRESENCE OF A "RECORD CONTAINS <INTEGER> TO <INTEGER>" CLAUSE WILL NOT, BY ITSELF, CAUSE THE FILE TO BE DECLARED WITH A NON-ZERO FILETYPE ATTRIBUTE. FILES HAVING A "BLOCK CONTAINS <INTEGER> TO <INTEGER>" CLAUSE WILL BE CONSIDERED AS VARIABLE LENGTH, BUT IF THE 01-LEVEL RECORD DESCRIPTIONS FOR THE FILE DO NOT SPECIFY A SIZE CLAUSE WHICH INDICATES A VARIABLE LENGTH ITEM THEN THE 01-LEVEL DATA ITEMS WILL

BE CONSIDERED TO BE FIXED LENGTH DATA ITEMS. THAT IS, THE VALUE OF THE FIRST FOUR CHARACTERS IN THE RECORD WILL ONLY BE USED BY LOGICAL I/O IN DETERMINING THE LENGTH OF A RECORD WRITTEN, NOT IN DETERMINING THE LENGTH OF THE 01-LEVEL ITEM.

IF THE "STACK" OPTION IS SET, THE COMPILE LISTING WILL INDICATE THE VALUES DETERMINED FOR THE BLOCKSIZE, MAXRECSIZE, MINRECSIZE, FILETYPE, AND INTMODE ATTRIBUTES.

D0710 COBOL - IMBEDDED QUOTES IN LITERALS - 01-20-74

DNOTE D0525 OF THE MARK 2.5 SYSTEM NOTES STATED THAT, BEGINNING ON THE MARK 2.6 RELEASE, COBOL WOULD ACCEPT IMBEDDED QUOTES WITHIN NON-NUMERIC LITERALS. EACH PAIR OF IMBEDDED CONTIGUOUS QUOTATION MARK CHARACTERS REPRESENTS A SINGLE QUOTATION MARK CHARACTER. FOR EXAMPLE: THE LITERAL """A""B""" WILL BE CONSIDERED TO BE THE FIVE CHARACTER STRING "A"B". THIS EXTENSION DOES NOT CONFLICT WITH THE USASI 1968 STANDARD, YET IS COMPATIBLE WITH BURROUGHS SMALL AND MEDIUM SYSTEMS COBOL LANGUAGES.

D0714 COBOL - EXAMINE WITH "LOW-VALUE" - 12-28-74

THE USE OF THE FIGURATIVE CONSTANTS "UPPER-BOUNDS" AND "LOWER-BOUNDS" IS NOW ALLOWED. PLEASE REFER TO COBOL D0725 FOR THE TABLE OF FIGURATIVE CONSTANT VALUES.

THIS CHANGE ALSO CORRECTS A PROBLEM IN WHICH THE USE OF THE FIGURATIVE CONSTANT "LOW-VALUE" IN AN EXAMINE STATEMENT CAUSED AN ALPHANUMERIC IDENTIFIER TO BE EXAMINED/REPLACED WITH SPACES, RATHER THAN THE "NULL" CHARACTER.

D0721 COBOL - USASI OPTION - COMP ITEMS - 12-06-73

WHEN THE "USASI" DOLLAR SYSTEM OPTION IS SET, ARITHMETIC VALUES BEING STORED INTO COMPUTATIONAL ITEMS ARE TRUNCATED TO THE NUMBER OF DECIMAL PLACES SPECIFIED BY THE PICTURE CLAUSE OF THE COMPUTATIONAL ITEM. ALSO, A POSITIVE VALUE IS STORED UNLESS THE PICTURE SPECIFIES THAT THE ITEM IS SIGNED.

D0722 COBOL - DOLLAR CARDS - 11-18-73

SEVERAL ADDITIONAL FACILITIES ARE AVAILABLE FOR COBOL DOLLAR CARDS.

THE STANDARD DOLLAR OPTIONS "OMIT", "LISTP", AND "LISTDELETED" HAVE BEEN IMPLEMENTED. THEIR MEANING IS DEFINED IN THE B6700 HANDBOOK.

USER DEFINED DOLLAR OPTIONS ARE ALSO NOW ALLOWED. IDENTIFIERS NOT HAVING THE SAME LENGTH AND FIRST FIVE CHARACTERS AS A STANDARD DOLLAR OPTION WILL BE CONSIDERED TO BE USER OPTIONS.

IN ADDITION, ANY OPTION MAY BE SET TO THE VALUE OF AN "OPTION EXPRESSION", AS IS THE CASE IN ESPOL AND ALGOL. THE SYNTAX FOR THIS CONDITIONAL SET IS:

SET <OPTION> = <OPTION EXPRESSION>

WHERE

<OPTION EXPRESSION>::= <OPTION SECONDARY> / <OPTION EXPRESSION>
<BOOLEAN OPERATOR><OPTION SECONDARY>
<OPTION SECONDARY>::= <OPTION PRIMARY> / NOT <OPTION PRIMARY>
<OPTION PRIMARY>::= <OPTION> / (<OPTION EXPRESSION>)
<OPTION>::= <STANDARD OPTION> / <USER OPTION>
<BOOLEAN OPERATOR>:= AND / OR / IMP / EQV

FOR EXAMPLE, IN THE FOLLOWING CARD DECK, CARDS IN REGION (1) WILL BE OMITTED AND CARDS IN REGION (2) WILL NOT BE OMITTED:

D0722 COBOL - DOLLAR CARDS - 11-18-73

```
$SET TESTING KLUDGE
    .
    .
    .
$SET OMIT=TESTING AND KLUDGE
    .
    .   (1)
    .
$POP OMIT
    .
    .
    .
$SET OMIT=NOT (TESTING OR KLUDGE)
    .
    .   (2)
    .
$POP OMIT
    .
    .
    .
```

THE SPECIAL-ACTION PARAMETER OPTIONS "AREACLASS", "FROM", "LEVEL", "LIMIT", AND "PAGE" ARE NOT PERMITTED IN AN OPTION EXPRESSION, SINCE THEY IMPLY NO BOOLEAN VALUE. THE NORMAL PRECEDENCE OF BOOLEAN OPERATORS APPLIES TO THE EVALUATION OF OPTION EXPRESSIONS. DOLLAR CARDS HAVING OPTIONS, BUT NO OPTION ACTION, WILL CAUSE ALL OPTIONS, BOTH STANDARD AND USER OPTIONS, TO BE CLEARED. FOR EXAMPLE, GIVEN THE USER OPTION TESTING, $TESTING WILL CAUSE THE SAME ACTION AS DESCRIBED IN THE COBOL REFERENCE MANUAL (5000656), PAGE 13-11 FOR AN OPTION WITH NO OPTION INDICATOR.

D0725 COBOL - CROSS-CLASS COMPARES AND MOVES - 12-28-73

THIS PATCH ALLOWS CROSS-CLASS COMPARISONS AND MOVES INVOLVING ALPHANUMERIC FIGURATIVE CONSTANTS AND NUMERIC ITEMS. THE LIST OF FIGURATIVE CONSTANTS IN THE COBOL MANUAL (5000656, PAGE 2-10)

SHOULD APPEAR AS FOLLOWS:

	TYPE	ZERO	SPACE	H-VAL	L-VAL	QUOTE	UPPER-BOUND	LOWER-BOUND	ALL
COMP-1	*	0	*	*	*	*	**	0	*
COMP	GROUP	0'S	*	*	*	*	*	0'S	*
	ELEM	0	*	*	*	*	**	0	*
COMP-2	GROUP	0 CHR	*	*	*	*	*	0 CHR	*
	ELEM	0 VAL	*	*	*	*	*	0 VAL	*
EBCD	AB	8"0"	8" "	4"FF"	4"00"	8"'"	8"Z"	8" "	
	AN	8"0"	8" "	4"FF"	4"00"	8"'"	4"FF"	4"00"	
	EA	8"0"	8" "	4"FF"	4"00"	8"'"	4"FF"	4"00"	
	NM	0 VAL	8" "	4"FF"	4"00"	8"'"	8"9"	0 VAL	
	EN	0 VAL	8" "	4"FF"	4"00"	8"'"	8"9"	0 VAL	
BCL	AB	6"0"	6" "	*	*	6"'"	6"Z"	6" "	
	AN	6"0"	6" "	*	*	6"'"	6"9"	6" "	
	EA	6"0"	6" "	*	*	6"'"	6"9"	6" "	
	NM	0 VAL	6" "	*	*	6"'"	6"9"	0 VAL	
	EN	0 VAL	6" "	*	*	6"'"	6"9"	0 VAL	
ASCII	AB	7"0"	7" "	4"FF"	4"00"	7"'"	7"Z"	7" "	
	AN	7"0"	7" "	4"FF"	4"00"	7"'"	4"FF"	4"00"	
	EA	7"0"	7" "	4"FF"	4"00"	7"'"	4"FF"	4"00"	

* INVALID

**MAX INTEGER VALVE, SINGLE OR DOUBLE

THIS PAGE INTENTIONALLY LEFT BLANK.

CONTROLLER

D0558 CONTROLLER - SHOW SWAP STATUS - 11-18-73

IF A TASK IS RUNNING IN A SUBSPACE THEN A <MIX> WHY WILL SHOW THE SWAP STATUS AS TO IN CORE, ON DISK, ETC.

D0562 CONTROLLER - CONTROLLER MESSAGE CHANGES - 11-18-73

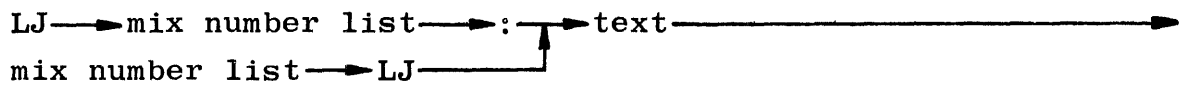
EP

THE CORRECT SYNTAX FOR THE EP MESSAGE IS "EP" ONLY. IF THE OPERATOR KEYS IN "EP <NUMBER>", THE MESSAGE "INVALID NUMBER" WILL BE DISPLAYED.

LC

THE LC COMMAND HAS BEEN RE-IMPLEMENTED.

LJ



THE LJ (LOG TO JOB LOG) MESSAGE MAY BE USED TO ENTER ANY COMMENT INTO THE APPROPRIATE JOB LOG(S) AND THE SYSTEM LOG (SYSTEM/SUMLOG).

EXAMPLE:

<MIX NUMBER> LJ HELLO JOB

LT

IT IS NOW POSSIBLE FOR THE OPERATOR TO LOAD A TRAIN PRINTER

WITH A PARTICULAR TRANSLATE TABLE. THE MESSAGE SYNTAX IS:

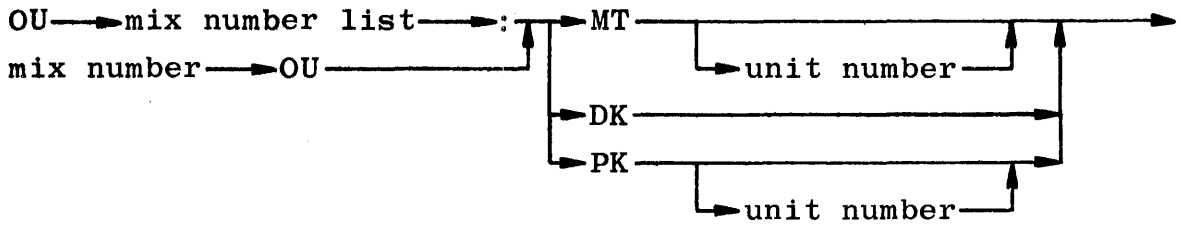
LT LP N EBCDIC 64

WHERE N IS THE UNIT NUMBER OF THE LINE PRINTER. THE SYSTEM WILL RESPOND, ON A SUCCESSFUL LOAD:

LP N EBCDIC 64 TRANSLATE TABLE LOADED.

OU

THE ABILITY TO DISPATCH OUTPUT TO DISK PACK HAS BEEN ADDED. THE SYNTAX IS AS FOLLOWS:



OUPK PLACES THE FILE ON THE APPROPRIATE DISK PACK. IF THE FILE HAD PACKNAME SET, THE CHOSEN DISK PACK IS THE PACK WITH THAT NAME. IF NO PACK NAME IS GIVEN, THE SYSTEM RESOURCE PACK IS USED.

OUPKNN PLACES THE FILE ON THE SPECIFIED PACK UNIT. IF NO PACKNAME HAS BEEN SPECIFIED BY THE PROGRAMMER, THE OPERATOR MAY DESIGNATE ANY NATIVE MODE WRITE-ENABLED BASE PACK. IF THE PACKNAME HAS BEEN SPECIFIED BY THE PROGRAMMER, ONLY A NATIVE MODE WRITE-ENABLED BASE PACK WITH THE SPECIFIED NAME MAY BE DESIGNATED.

RO, SO, TO OPTION ADDITION

TWO NEW RUN-TIME SYSTEM OPTIONS, FULLTRANSLATION AND BACKUPBYJOBNR, HAVE BEEN ADDED.

FULLTRANSLATION (OPTION 18)

SETTING FULLTRANSLATION CAUSES EVERY LOGICAL FILE TO BE INITIALIZED WITH THE FILE ATTRIBUTE TRANSLATE SET TO

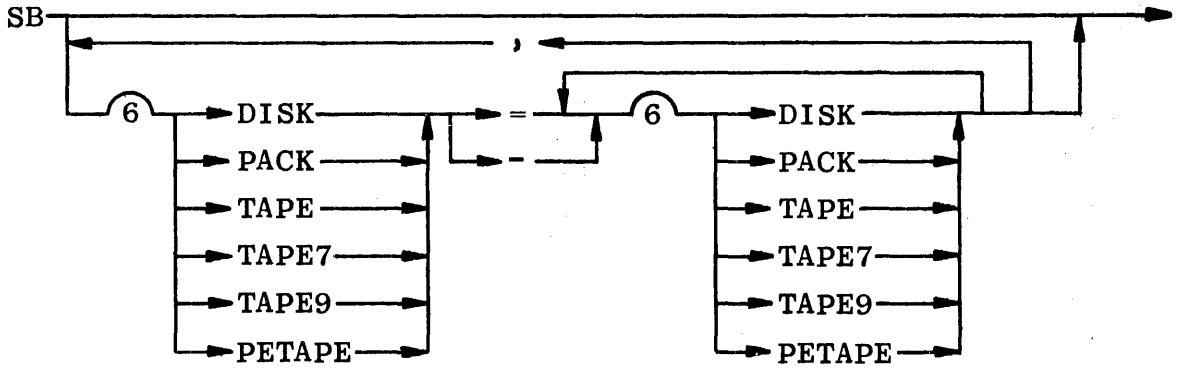
FULLTRANS. THIS ALLOWS A CONSISTENT USE OF SOFTWARE TRANSLATION (WHEN HARDWARE TRANSLATION IS NOT PROVIDED) WITHOUT THE RESTRICTIONS WHICH WERE APPLIED BECAUSE OF INCONSISTENCIES IN THE 2.2 AND PREVIOUS MCP"S. SEE THE DISCUSSION OF SOFTWARE TRANSLATION IN THE INPUT/OUTPUT SECTION OF THE SYSTEM NOTES.

BACKUPBYJOBNR (OPTION 17)

WHEN BACKUPBYJOBNR IS SET, JOBS ARE PRINTED BY ORDER OF THE JOB NUMBER. WHEN RESET, JOBS ARE PRINTED IN REVERSE ORDER OF PRINT QUANTITY.

SB

THE SB COMMAND ENABLES THE OPERATOR TO SUBSTITUTE ONE BACKUP MEDIUM FOR ANOTHER. BELOW IS THE SYNTAX TO BE USED:



NOTE: THE FOLLOWING SYNONYMS MAY BE USED:

DISK = DK
PACK = PK
TAPE = MT
TAPE7 = MT7
TAPE9 = MT9
PETAPE = MTP

FOR EXAMPLE:

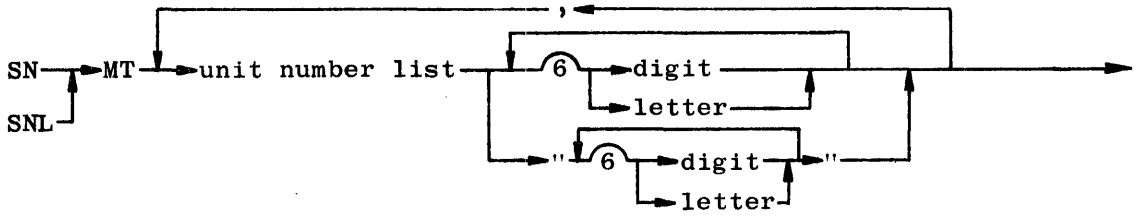
SB DISK=PACK

THIS SB COMMAND WILL CAUSE ALL BACKUP TO DISK TO GO TO PACK.

PLEASE REFER TO THE SYSTEM MISCELLANEA FOR FURTHER INFORMATION ON THE SB INPUT COMMAND.

SN AND SNL

THE SN SYNTAX IN THE WORK FLOW MANAGEMENT USERS GUIDE SHOULD BE CHANGED TO:



THE SN (SERIAL NUMBER) MESSAGE IS USED TO PURGE (WITH OPTIONAL LOCKING) AND ASSIGN A SERIAL "NUMBER" TO A TAPE UNIT. THIS "NUMBER" MAY CONSIST OF UP TO SIX ALPHANUMERIC CHARACTERS. IF A NUMBER IS USED IT WILL BE RIGHT JUSTIFIED WITH LEADING ZEROES ADDED. ANY "NUMBER" CONTAINING ALPHABETIC CHARACTERS, OR ANY QUOTED ALPHANUMERIC STRING WILL BE LEFT JUSTIFIED WITH TRAILING BLANKS.

WHEN THE SNL FORM OF THE MESSAGE IS USED, THE TAPE IS LOCKED AS WELL AS PURGED SO NO JOB WILL AUTOMATICALLY PICK UP THE SCRATCHED TAPE.

EXAMPLE:

```
SN MT  1 1, 2 "2", 3-5 TEST, 6 "ABC"
RESULT:  MT      SERIAL
         1       [000001]
         2       [2     ]
         3       [TEST  ]
         4       [TEST  ]
         5       [TEST  ]
         6       [ABC   ]
```

TERM USER <USERCODE>/<PASSWORD>

D0562 CONTROLLER - CONTROLLER MESSAGE CHANGES - 11-18-73

A NEW TERM ATTRIBUTE "USER" MAY BE SET ON THE OPERATOR TERMINAL. WHEN SET ALL PD REQUESTS AND CONTROLCARDS WILL BE HANDLED UNDER THAT USERCODE. TO CANCEL THE USERCODE USE: TERM USER-.

D0711 CONTROLLER - DISPLAY OF STATUS CHANGES - 02-20-74

THE FOLLOWING CHANGES HAVE BEEN MADE TO DISPLAYS OF STATUS:

MIX TABLE

RJE JOB INDICATION

JOBS RUN UNDER RJE WILL BE PRECEDED BY THE TIMES SIGN "x".

PERIPHERAL TABLE

CYCLE AND VERSION VALUES OF MAGNETIC TAPES

THE CYCLE AND VERSION VALUES OF MOUNTED MAGNETIC TAPES WILL BE DISPLAYED IN RESPONSE TO AN OL OR PER MT <UNIT NUMBER> SYSTEM INPUT MESSAGE IF THEIR VALUES ARE OTHER THAN THE DEFAULT VALUES. THIS INFORMATION WILL FOLLOW THE REEL NUMBER WITH THE CYCLE NUMBER FIRST FOLLOWED BY A COLON THEN THE VERSION NUMBER, FOR EXAMPLE:

MT 17*7[000017] #1 2:15 RMC

D0718 CONTROLLER - REMOTESPO CAPABILITIES - 03-03-74

REMOTESPO CAPABILITIES ARE ONLY APPLICABLE TO TELETYPES, CONRACS, OR TD800"S.

DATACOM

D0554 MCP-DATACM - DCP CLUSTER FAILURE - 12-28-73

IF A CLUSTER FAILS WHEN DCP INITIALIZATION OCCURS, NO INDICATION WAS PREVIOUSLY GIVEN THAT THE INITIALIZATION REQUEST WAS ONLY PARTIALLY SUCCESSFUL. WITH THIS CHANGE, THE MESSAGE "CLUSTER #N FAILED", WHERE N IS THE NUMBER OF THE FAILING CLUSTER, IS DISPLAYED FOR EACH CLUSTER THAT IS NOT ON LINE AND READY AT THE TIME OF INITIALIZATION. NOTE: THE "HOLD P-L/STOP ON FAULT" SWITCH ON THE DCP MAINTENANCE DISPLAY PANEL MUST BE IN THE NEUTRAL (RUN) POSITION; OTHERWISE, THE DCP WILL HALT WHEN THE CLUSTER FAULT OCCURS.

DCP CODE FILES GENERATED BY 2.5 OR EARLIER NDL COMPILERS MAY CAUSE ERRONEOUS "CLUSTER FAILED" MESSAGES. IF THIS OCCURS, RECOMPILE USING THE 2.6 NDL COMPILER.

D0563 MCP-DATACM - DEFAULT POLLING FREQUENCY - 11-18-73

THE ENABLEINPUT DCWRITE CONTAINS IN THE VARIANT FIELD (MSG[0].[31:8]) THE VALUE TO BE STORED INTO THE POLLING FREQUENCY BYTE OF A STATION, "STATION(FREQUENCY)". AN MCS CAN NOW ELECT TO USE THE STATION FREQUENCY DEFAULT VALUE AS SPECIFIED IN NDL BY SETTING THE VARIANT FIELD TO 255.

D0579 MCP-DATACM - INTERROGATE STA ENVIR DCWRITE - 11-18-73

THE "INTERROGATE STATION ENVIRONMENT" DCWRITE FUNCTION ENABLES AN MCS TO GAIN ACCESS TO MCP INFORMATION RELATED TO ONE OR MORE DATACOM STATIONS. THE EXTENSION OF THIS DCWRITE FUNCTION ALLOWS AN MCS TO ALSO OBTAIN DCP INFORMATION CONCERNING THE PARTICULAR

STATION(S) IN QUESTION. IF THIS IS DESIRED, THE FOLLOWING BITS IN THE MESSAGE VARIANT ARE TO BE USED:

MSG[0].[30:1] = 1 => DCP STATION INFORMATION DESIRED
MSG[0].[29:1] = 1 => DCP LINE INFORMATION DESIRED

NOTE:

THE SETTING OF THESE BITS WILL ONLY BE EFFECTIVE IF THE EQUIVALENT BITS FOR MCP INFORMATION ARE ALSO SET, I.E., TO OBTAIN STATION INFORMATION OF THE DCP, MSG[0].[30:1] AND MSG[0].[25:1] HAVE TO BE SET; TO OBTAIN DCP LINE INFORMATION, MSG[0].[29:1] AND MSG[0].[27:1] HAVE TO BE SET.

THE REQUEST FOR DCP INFORMATION RESULTS IN THE EXTENDED FORMAT OF THE INTERROGATE STATION ENVIRONMENT RESULT:

1. DCP STATION INFORMATION

THE THIRD WORD OF STATION INFORMATION IN THE EXTENDED FORMAT NOW CONTAINS THE STATION DESCRIPTOR OF THE DCP, FOLLOWED BY SIX OR EIGHT WORDS OF THE STATION TABLE OF THE DCP. THESE WORDS ARE INTERPRETED AS FOLLOWS:

A. THIRD WORD (MSG[MSG[INX1].[15:8]+2])
(WHERE INX1 = MSG[6].[7:8])

47:8 STATION PRIORITY
39:1 STATION ACKNOWLEDGE
31:8 STATION TALLY
23:1 STATION IS QUEUED
22:1 STATION ENABLED
21:1 STATION IS NOT READY
20:1 STATION IS INVALID

B. FOURTH WORD (MSG[MSG[INX1].[15:8]+3])

47:8 STATION CONTROL CHARACTER
39:8 STATION END OF TEXT CHARACTER
31:8 STATION BACKSPACE CHARACTER
23:8 STATION LINE DELETE CHARACTER

15:8 STATION WRU (WHO ARE YOU) CHARACTER

C. FIFTH WORD (MSG[MSG[INX1].[15:8]+4])

47:48 STATION QUEUE

D. SIXTH WORD (MSG[MSG[INX1].[15:8]+5])

47:8 STATION AVAILABLE SPACE INDEX
31:8 LAST FLAG SET IN NDL
23:24 NDL STATION FLAGS

E. SEVENTH WORD (MSG[INX1].[15:8]+6])

47:48 STATION TRANSMISSION NUMBER

F. EIGHTH WORD (MSG[MSG[INX1].[15:8]+7])

47:24 RECEIVE ADDRESS CHARACTERS
23:24 TRANSMIT ADDRESS CHARACTERS

G. NINTH WORD (MSG[MSG[INX1].[15:8]+8])

47:8 STATION TOGGLES
39:8 CURRENT NUMBER OF STATION RETRIES LEFT
31:8 INITIAL NUMBER OF STATION RETRIES
23:8 STATION TALLY[2]
15:8 STATION TALLY[1]
7:8 STATION TALLY[0]

THE TENTH AND ELEVENTH WORDS WILL CONTAIN SEQUENCE MODE INFORMATION, AND WILL EXIST ONLY IF SEQUENCE MODE EXISTS FOR THE STATION.

2. DCP LINE INFORMATION

THE SECOND WORD OF LINE INFORMATION IN THE EXTENDED FORMAT NOW CONTAINS THE LINE DESCRIPTOR OF THE DCP, FOLLOWED BY TWO WORDS OF THE LINE TABLE OF THE DCP. THESE WORDS ARE INTERPRETED AS FOLLOWS:

A. SECOND WORD (MSG[MSG[INX1].[31:8]+1])

47:1 LINE IS PRIMARY (FULL DUPLEX ONLY)
46:1 LINE IS A SECONDARY (FULL DUPLEX ONLY)
45:6 LINE CONTROL INDEX
31:1 LINE NOT READY PENDING
30:1 SWITCHED LINE ERROR
29:1 PHONE RINGING
28:1 BUSY IN SWITCHED REQUEST
27:1 LINE IS CONNECTED
26:1 AUTO ANSWER IN FORCE
25:1 LINE HAS ASSOCIATED ACU (AUTOMATIC CALLING UNIT)
24:1 LINE IS A SWITCHED LINE
23:1 LINE HAS CONTROLLED CARRIER
22:1 LINE IS BUSY
21:1 LINE NOT TO BE USED
20:1 SYNCHRONOUS LINE

B. THIRD WORD (MSG[MSG[INX1].[31:8]+2])

47:1 LINE NOT READY
46:1 LINE BUSY
45:1 LINE WRITE READY
44:1 LINE ACKNOWLEDGE READY
43:1 LINE NOT CONNECTED
42:1 LINE IS QUEUED
41:1 LINE TOGGLE ONE
40:1 LINE TOGGLE ZERO
39:8 CURRENT STATION NUMBER
31:8 MAXIMUM NUMBER OF STATIONS ON THIS LINE
23:8 LINE TALLY[1]
15:8 LINE TALLY[0]
7:8 INDEX FOR COLINE (FULL DUPLEX ONLY)

C. FOURTH WORD (MSG[MSG[INX1].[31:8]+3])

47:8 ADAPTER TYPE FIELD
39:8 INITIATE TRANSMIT DELAY
31:8 INITIATE RECEIVE DELAY
23:8 TIMEOUT VALUE FOR THIS LINE

15:8 INDEX FOR ASSOCIATED ACU
7:8 INDEX FOR CONTINUE (FULL DUPLEX ONLY)

D0582 MCP-DATACM - DIAGNOSTIC DCWRITE FUNCTIONS - 12-06-73

A NEW SERIES OF DCWRITE FUNCTIONS HAVE BEEN IMPLEMENTED TO ALLOW LIMITED DCP DIAGNOSTIC FUNCTIONS. THESE FUNCTIONS ALLOW AN MCS TO PERFORM ON-LINE TEST OF LINE ADAPTORS AND CLUSTERS ON READY AND INITIALIZED DCPS.

THESE DCWRITES MAY ONLY BE PERFORMED BY AN MCS WHICH HAS BEEN DECLARED IN NDL AS "CONTROL=TRUE" (SEE 2.5 SYSTEM NOTE D0350). THE MINIMUM SIZE FOR THESE DCWRITES IS EIGHT WORDS.

MAINTENANCE TRANSFER RESULT (TYPE = 22)

FORMAT OF MESSAGE:

```
MSG[0].[47:08] = 22
      .[39:16] = 0 => STATION IS BEING DETACHED
                 1 => STATION IS BEING ATTACHED
      .[23:24] = LSN

MSG[1]         = NUMBER OF DIAGNOSTIC MCS
```

SEMANTICS:

FOR EACH STATION WHICH IS BEING TRANSFERED TO/FROM THE NDL-DEFINED (PRIMARY) MCS AND A DIAGNOSTIC MCS, A "MAINTENANCE TRANSFER" RESULT MESSAGE IS GENERATED. EACH MESSAGE IS INSERTED INTO THE PRIMARY QUEUE OF THE PRIMARY MCS. IF THE TRANSFER IS FROM THE PRIMARY MCS TO THE DIAGNOSTIC MCS, THE VARIANT FIELD OF THE MESSAGE (MSG[0].[39:16]) WILL BE ONE, OTHERWISE ZERO. WHILE A STATION IS BEING CONTROLLED BY THE DIAGNOSTIC MCS, THE PRIMARY MCS WILL NOT BE ALLOWED TO PERFORM ANY DCWRITE REQUESTS TO THE STATION. NO SPECIAL ACTION IS REQUIRED BY THE PRIMARY MCS WHEN THIS MESSAGE IS RECEIVED.

ATTACH LINE ADAPTOR (TYPE = 160)

FORMAT OF MESSAGE PARAMETER:

```
MSG[0].[47:08] = 160
      .[23:01] = 1
      .[22:01] = DCP NUMBER
      .[15:04] = CLUSTER NUMBER
      .[11:04] = ADAPTOR NUMBER
```

SEMANTICS:

BEFORE ANY DIAGNOSTIC FUNCTIONS CAN BE PERFORMED ON A LINE, THE DIAGNOSTIC MCS MUST BE IN CONTROL OF THE LINE TO BE TESTED, AND NO OTHER ACTIVITY SHOULD BE ALLOWED FOR THAT LINE WHILE DIAGNOSTICS ARE IN PROGRESS. THIS DCWRITE ALLOWS AN MCS TO ATTACH ITSELF TO A LINE WITHOUT THE COOPERATION OF THE MCS CURRENTLY IN CONTROL OF THE LINE.

FOR EACH STATION WHICH EXISTS ON THE LINE TO BE ATTACHED, A "MAINTENANCE TRANSFER" RESULT MESSAGE (TYPE=22) IS PLACED IN THE PRIMARY QUEUE OF THE CURRENT OWNER OF THE STATION. THE PRIMARY QUEUE OF EACH STATION IS CHANGED TO THE PRIMARY QUEUE OF THE REQUESTING MCS. IF THE LINE IS ALREADY CONTROLLED BY ANOTHER DIAGNOSTIC MCS, DCWRITE WILL RETURN ERROR #125 (LINE ALREADY ATTACHED).

EXAMPLE:

```
ALLOCATE(MSG,8);
MSG[0]:= 0 & (160) [47:8]
         & (1) [23:1] & (DCPNR) [22:7]
         & (LINENR) [15:8];
RESULT:= DCWRITE(MSG);
```

DETACH LINE ADAPTOR (TYPE = 161)

FORMAT OF MESSAGE:

```
MSG[0].[47:08] = 161
```

.[23:01] = 1
.[22:07] = DCP NUMBER
.[15:04] = CLUSTER NUMBER
.[11:04] = ADAPTOR NUMBER

SEMANTICS:

THIS DCWRITE CAUSES ALL STATIONS ON THE SPECIFIED LINE TO BE RETURNED TO THE CONTROL OF THEIR ORIGINAL MCS. FOR EACH STATION ON THE LINE A "MAINTENANCE TRANSFER" RESULT MESSAGE (TYPE=22) IS PLACED IN THE PRIMARY QUEUE OF THE ORIGINAL OWNER OF THE STATION. IN ADDITION, THE PRIMARY AND CURRENT QUEUES FOR EACH STATION ARE SET TO THE PRIMARY QUEUE OF THE ORIGINAL OWNER. IF THE LINE HAS NOT BEEN ATTACHED, DCWRITE WILL RETURN ERROR #126 (LINE NOT ATTACHED).

EXAMPLE:

```
ALLOCATE(MSG,8);
MSG[0]:= 0 & (161) [47:8] & DL[23:24];
RESULT:= DCWRITE(MSG);
```

ATTACH CLUSTER (TYPE = 162)

FORMAT OF MESSAGE:

SAME AS FOR "ATTACH LINE ADAPTOR" (TYPE=160).

SEMANTICS:

SINCE SOME DIAGNOSTIC TESTS INVOLVE AN ENTIRE CLUSTER, THIS DCWRITE ALLOWS UP TO 16 LINE ADAPTORS TO BE ATTACHED AT ONE TIME. MSG[0].[11:04] IS NOT USED. INSTEAD, ALL VALID ADAPTORS ON THE CLUSTER SPECIFIED BY MSG[0].[15:04] ARE ATTACHED. THIS DCWRITE FUNCTIONS SIMILAR TO THE "ATTACH LINE ADAPTOR" DCWRITE. IF THE SPECIFIED CLUSTER IS INVALID, DCWRITE RETURNS ERROR #124 (UNKNOWN CLUSTER).

EXAMPLE:

```
ALLOCATE(MSG,8);
```

```
      MSG[0]:= 0 & (162) [47:8] & (1) [23:1]
                 & (DCPNR) [22:7] & (CLUSTER) [15:4];
      RESULT:= DCWRITE(MSG);
```

DETACH CLUSTER (TYPE = 163)

FORMAT OF MESSAGE:

SAME AS FOR "DETACH LINE ADAPTOR" (TYPE = 161).

SEMANTICS:

SIMILAR TO "DETACH LINE ADAPTOR" EXCEPT IT ALLOWS DETACHMENT OF ALL 16 LINES ON A CLUSTER.

EXAMPLE:

```
      ALLOCATE(MSG,8);
      MSG[0]:= 0 & (163) [47:8] & (1) [23:1]
                 & (DC) [22:23];
      RESULT:= DCWRITE(MSG);
```

ADAPTOR WRITE (TYPE = 166)

FORMAT OF MESSAGE:

```
      MSG[0].[47:8] = 166
            .[39:8] = VALUE FOR AC REGISTER
            .[31:8] = VALUE FOR AI REGISTER
            .[23:1] = 1
            .[22:7] = DCP NUMBER
            .[15:4] = CLUSTER NUMBER
            .[11:4] = ADAPTOR NUMBER
```

SEMANTICS:

THIS DCWRITE CAUSES THE DCP TO LOAD THE AC AND AI REGISTERS WITH THE SPECIFIED VALUES, AND THEN PERFORM AN ADAPTOR WRITE (AWI) INSTRUCTION TO THE SPECIFIED ADAPTOR AND CLUSTER.

NOTE: THIS DCWRITE ALWAYS LEAVES THE LINE NOT READY AND NOT BUSY.

EXAMPLE:

```
ALLOCATE(MSG,8);
MSG[0]:= 0 & (166) [47:8] & (1) [23:1]
          & (DL) [22:15]
          & (1) [39:8]            %AC=1 (C2)
          & (4"6F") [31:8]        %AI=4"6F"
          ;

RESULT:= DCWRITE(MSG);
```

THE ABOVE DCWRITE WOULD STORE 4"6F" IN THE C2 REGISTER.

ADAPTOR INTERROGATE (TYPE = 167)

FORMAT OF MESSAGE:

```
MSG[0].[47:8] = 167
      .[39:8] = VALUE FOR AC REGISTER
      .[23:1] = 1
      .[22:7] = DCP NUMBER
      .[15:4] = CLUSTER NUMBER
      .[11:4] = ADAPTOR NUMBER
```

SEMANTICS:

THIS DCWRITE CAUSES THE DCP TO LOAD THE AC REGISTER WITH THE VALUE SPECIFIED IN MSG[0].[39:8], AND THEN EXECUTE AN ADAPTER INTERROGATE (AWRR) INSTRUCTION TO THE SPECIFIED ADAPTOR AND CLUSTER.

THE RESULTING VALUES OF AC AND AI ARE RETURNED IN THE "GOOD RESULTS" (TYPE=5) MESSAGE:

```
MSG[0].[39:8] = AC REGISTER
      .[31:8] = AI REGISTER.
```

EXAMPLE:

```
ALLOCATE(MSG,8);
MSG[0]:= 0 & (167) [47:8] & (1) [23:1]
```

D0582 MCP-DATACM - DIAGNOSTIC DCWRITE FUNCTIONS - 12-06-73

```
                & (DL) [22:15];
      RESULT:= DCWRITE(MSG);
```

THE ABOVE DCWRITE WOULD CAUSE THE VALUE OF THE ADAPTOR TYPE FIELD TO BE RETURNED IN MSG[0].[31:8].

D0598 MCP-DATACM - ALLOW DLS FOR DCWRITE - 12-28-73

STATION ORIENTED DCWRITES PREVIOUSLY REQUIRED AN LSN IN MSG[0].[22:23]. THIS EXTENSION ALLOWS A DLS TO BE USED INSTEAD, INDICATED BY MSG[0].[23:1]=1. THE DCP NUMBER IS EXPECTED TO BE FOUND IN MSG[0].[22:7], LINE NUMBER IN MSG[0].[15:8] AND STATION NUMBER IN MSG[0].[7:8].

D0599 MCP-DATACM - SOURCENDL VINTAGE - 12-28-73

INITIALIZE PRIMARY QUEUE NOW RETURNS THE VINTAGE OF SOURCENDL, WHICH THE CURRENT DATACOM FILES HAVE BEEN COMPILED WITH, IN MSG[3]. THIS WORD IS DEFINED AS FOLLOWS:

```
MSG[3].[35:12] = MARKLEVEL
MSG[3].[23:12] = CYCLE NUMBER
MSG[3].[11:12] = PATCH NUMBER
```

D0600 MCP-DATACM - NIFPREFIX SIZE - 12-28-73

AN INITIALIZE PRIMARY QUEUE DCWRITE RETURNS THE DATACOM <FILE PREFIX> IN EXTERNAL STANDARD FORM STARTING IN MSG[6]. ITS LENGTH IN TERMS OF NUMBER OF CHARACTERS IS NOW STORED IN THE TEXTSIZE FIELD (MSG[2].[39:16]).

D0601 MCP-DATACM - DCP RESULTS ON EMPTY LINE - 12-28-73

D0601 MCP-DATACM - DCP RESULTS ON EMPTY LINE - 12-28-73

IF A DCP RESULT WAS GENERATED FOR A LINE WITH NO STATIONS (SUCH AS AN ACU OR AUXILIARY FULL DUPLEX LINE), THE DCC WOULD FAULT ATTEMPTING TO PROCESS THE MESSAGE. WITH THIS CHANGE, THESE RESULT MESSAGES ARE FORWARDED TO THE PRIMARY QUEUE OF THE MCS WHICH GENERATED THE ORIGINAL DCP REQUEST.

D0602 MCP-DATACM - MCS HANDLING - 12-28-73

THE FORMAT OF THE MCS TABLE HAS BEEN UPDATED. IN THE DCALGOL REFERENCE MANUAL, APPENDIX B, PAGE B-8, THE FOLLOWING ADDITIONAL FIELD DEFINITIONS SHOULD BE ADDED TO PARAGRAPH D:

```
[47:1] = 1 => MCS HAS DIAGNOSTIC CAPABILITIES
[44:1] = 1 => "MCS REQUIRED" MESSAGE HAS BEEN DISPLAYED FOR
              THIS MCS
```

D0716 MCP-DATACM - NDL FILES BEFORE 2.6 - 02-20-74

AS OF THE 2.7 SYSTEM RELEASE, NDL FILES GENERATED BEFORE 2.6 WILL NOT BE ACCEPTED. ALL NDL CODE SHOULD BE RECOMPILED USING THE 2.6 NDL COMPILER BEFORE THE 2.7 RELEASE.

DATA MANAGEMENT

D0719 DM6700 - RECONSTRUCTION CHANGES - 02-20-74

SEVERAL NEW STATUS VALUES MAY BE RETURNED WHEN RUNNING RECONSTRUCTION:

VALUE	MEANING
26	THE STARTING AUDIT FILE WAS NOT SPECIFIED.
27	RECONSTURCTION DOES NOT HAVE ENOUGH BUFFERS. (FORMERLY, THE STATUS RETURNED IN THIS CASE WAS 25, WHICH COULD ALSO MEAN THAT AN IRRECOVERABLE I/O ERROR HAD OCCURRED.)
28	THE STARTING AUDIT FILE WAS NOT FOUND IN THE AUDIT ARCHIVE.
29	THE STARTING AUDIT FILE WAS CREATED TOO LATE (AFTER THE FIRST ERROR ON THE FILE TO BE RECONSTRUCTED).

FOR RECONSTRUCTION TO BE SUCCESSFUL, IT MUST CAPTURE ALL THE INFORMATION IN THE ROWS BEING RECONSTRUCTED. THUS, EVERY RECORD MUST EITHER APPEAR IN THE AUDIT OR BE ALREADY CORRECT IN THE BACKUP DUMP. THE DUMP MUST BE TERMINATED BEFORE THE FIRST ERROR OCCURRED. THE USER MUST CHECK THIS BY PRINTING THE ERRORS FILE USING DMPRINTIT, AND COMPARING THE DATE AND TIME OF THE FIRST ERROR TO THE DATE AND TIME THE DUMP WAS COMPLETED.

IF THE DUMP WAS TAKEN WHILE THE DATA BASA WAS SHUT DOWN, THE STARTING AUDIT FILE CAN BE THE NEXT AUDIT FILE CREATED AFTER THE DUMP OR ANY EARLIER AUDIT FILE. THE USER CAN DETERMINE THIS BY PRNTING THE AUDIT ARCHIVE WITH DMPRINTIT.

IF THE DUMP WAS TAKEN WHILE THE DATA BASE WAS IN USE, SPECIAL CARE MUST BE TAKEN TO INSURE THAT THE STARTING AUDIT FILE WAS CREATED

EARLY ENOUGH. BECAUSE AN ON-LINE DUMP WILL NOT DUMP ANY INFORMATION CURRENTLY IN THE BUFFERS OF THE MONITOR, IMAGES FOR ALL SUCH RECORDS MUST BE SEEN IN THE AUDIT. THE MONITOR PERIODICALLY FLUSHES ALL THE BUFFERS THAT WERE CHANGED BUT ARE CURRENTLY UNLOCKED AND WRITES A CONTROL RECORD ON THE AUDIT. IF INFORMATION ENTERS THE AUDIT AT A POINT SUCH THAT MANY OF THESE CONTROL POINTS OCCURRED PRIOR TO THE START OF THE ON-LINE DUMP, THEN ALL NECESSARY AFTER IMAGES WILL BE SEEN. AS A RULE OF THUMB, THE USER SHOULD SPECIFY THE STARTING AUDIT FILE TO BE THE ONE PRIOR TO THE AUDIT FILE THAT WAS CURRENT WHEN THE FIRST ERROR OCCURRED, OR ANY AUDIT FILE EARLIER THAN THAT ONE. THIS MAY BE DETERMINED BY PRINTING THE AUDIT ARCHIVE WITH DMPRINTIT. BECAUSE THE MONITOR HAS NO KNOWLEDGE OF WHEN THE BACKUP DUMP WAS CREATED, THE USER MUST SPECIFY THE STARTING SERIAL NUMBER FOR RECONSTRUCTION; OTHERWISE, A STATUS OF 26 WILL BE RETURNED.

DCALGOL

D0699 DCALGOL - RJE ATTRIBUTES TO DCALGOL - 10-28-73

FIVE NEW TASK ATTRIBUTES HAVE BEEN ADDED TO DCALGOL.

1. DESTNAME - THIS IS A STRING VALUED ATTRIBUTE WHICH MAY BE SET TO ANY STATION NAME DEFINED IN THE NETWORK DEFINITION LANGUAGE (NDL) OR THE LITERAL "SITE.". SETTING THIS ATTRIBUTE TO A STATION NAME WILL CAUSE ALL PRINTER AND PUNCH BACKUP TO BE BUILT UNDER THE DIRECTORIES "REMLPXX" OR REMCPXX" RESPECTIVELY. THE "XX" IN THE TITLES WILL BE THE MCS NUMBER ASSIGNED TO THAT STATION.

   THIS ATTRIBUTE MAY ALSO BE READ, IN WHICH CASE IT WILL RETURN THE STATION NAME ASSOCIATED WITH THE DESTINATION UNIT. IF INTERROGATED AND A REMOTE DESTINATION HAS NOT BEEN SPECIFIED, IT WILL RETURN THE STRING "SITE.".

2. DESTSTATION - THIS IS AN INTEGER VALUED ATTRIBUTE WHICH SETS OR RETURNS THE DESTINATION STATION. THIS ATTRIBUTE MAY ONLY BE SET BY AN MCS; AND WHEN SET, THE DESTINATION MCS NUMBER WILL BE SET USING THE NUMBER ASSOCIATED WITH THE MCS SETTING THIS ATTRIBUTE.

3. SOURCESTATION - THIS IS AN INTEGER VALUED ATTRIBUTE WHICH SETS OR RETURNS THE ORIGINATION STATION (LSN). THIS ATTRIBUTE MAY ONLY BE SET BY AN MCS.

4. SOURCEKIND - THIS IS AN INTEGER VALUED ATTRIBUTE WHICH RETURNS THE UNIT TYPE ASSOCIATED WITH THE ORIGINATING UNIT. FOR EXAMPLE:

   SPO=2, REMOTE=3, CARDREADER=9

5. MAXWAIT - THIS IS A REAL VALUED ATTRIBUTE WHICH SPECIFIES THE MAXIMUM NUMBER OF SECONDS A TASK CAN AFFORD TO WAIT ON

SPECIFIED SYSTEM FUNCTIONS. THOSE FUNCTIONS WHICH USE THIS VALUE WILL BE NOTED IN THE SYSTEM NOTES.

NEW QUEUE ATTRIBUTE:

QREMOVEWAIT - THIS IS A BOOLEAN VALUED ATTRIBUTE WHICH WHEN SET TRUE WILL CAUSE A REMOVE ON AN EMPTY QUEUE TO WAIT UNTIL A MESSAGE IS INSERTED. THIS FUNCTIONS DIFFERENTLY FROM A HOLD IN THAT IF THE REMOVING PROCESS IS RUNNING IN A SWAP SPACE, IT WILL BE SWAPPED OUT.

MISCELLANEOUS NEW CONSTRUCTS:

1. WFLCOMPILER - THIS IS A RECOGNIZED PROCEDURE WHICH EXPECTS TWO PARAMETERS. THE FIRST PARAMETER IS A QUEUE AND THE SECOND IS A BOOLEAN EXPRESSION. THIS PROVIDES A MECHANISM FOR A PROCESS TO INITIATE THE WORK FLOW LANGUAGE (WFL) COMPILER PASSING CARD IMAGES IN THE QUEUE. THE SECOND PARAMETER IF TRUE WILL REQUIRE THAT ALL JOBS SPECIFY A USERCODE. FOR EXAMPLE:

```
QUEUE Q;
TASK T;
PROCESS WFLCOMPILER(Q,TRUE) [T];
```

QUEUE ARRAYS HAVE BEEN EXTENDED TO ALLOW QUEUE REFERENCE ARRAYS. THESE ARE SYNTACTICALLY EQUIVALENT TO NORMAL ARRAY REFERENCE DECLARATIONS. AN EXAMPLE OF THEIR SYNTAX IS SHOWN BELOW.

2. SETUPINTERCOM - THIS IS A RECOGNIZED PROCEDURE WHICH EXPECTS TWO PARAMETERS. THE FIRST PARAMETER IS A QUEUE ARRAY REFERENCE AND THE SECOND IS A QUEUE. THIS PROVIDES A MEANS FOR AN MCS TO COMMUNICATE WITH OTHER MCS"S OR THE SYSTEM CONTROLLER. ONCE THIS PROCEDURE CALL HAS BEEN INVOKED THE PROCESS MAY SEND A MESSAGE TO MCS NUMBER N BY INSERTING A MESSAGE INTO THE QUEUE ARRAY REFERENCE INDEXED BY N. RECEIVED MESSAGES WILL BE FOUND IN THE QUEUE SPECIFIED BY THE SECOND PARAMETER TO SETUPINTERCOM. FOR EXAMPLE:

```
QUEUE ARRAY REFERENCE
   INTERCOMQUEUES[0];
QUEUE RECEIVE;
SETUPINTERCOM(INTERCOMQUEUES,RECEIVE);
```

THE SYSTEM CONTROLLER HAS MCS NUMBER ZERO. WHEN COMMUNICATING WITH THE CONTROLLER THE MESSAGES MUST BE STRUCTURED AS DESCRIBED IN THE REMOTE JOB ENTRY DOCUMENT.

DCALGOLINTRINSICS (DCALGOLINT)

D0603 DCALGOLINT - DCERRANALYSIS FUNCTION - 12-28-73

THE DOCUMENTATION FOR THE DCERRANALYSIS FUNCTION WILL BE FOUND IN THE SYSTEM MISCELLANEA.

DCPPROGEN

D0604 DCPPROGEN - DIAGNOSTIC MCS REQUESTS - 12-28-73

THE DCP WILL PERFORM AN ADAPTOR WRITE (AWI) OR AN ADAPTOR INTERROGATE (AWRR) WHEN AN MCS, WHICH IS DECLARED AS "CONTROL = TRUE", REQUESTS THIS FUNCTION. FOR FURTHER INFORMATION REFER TO DATACOM D0582.

D0605 DCPPROGEN - "INITIATE BREAK" STATEMENT - 12-28-73

THE USER CAN LEAVE AN ADAPTOR IN A BREAK STATE UNTIL A SUBSEQUENT ADAPTOR REQUEST. THE SYNTAX FOR THIS REQUEST SECTION STATEMENT IS:

<INITIATE BREAK STMT>::= INITIATE BREAK

THIS STATEMENT IS DESIGNED PRIMARILY FOR REVERSE CHANNEL APPLICATIONS.

D0606 DCPPROGEN - "AI" ADDED AS BYTE VARIABLE - 12-28-73

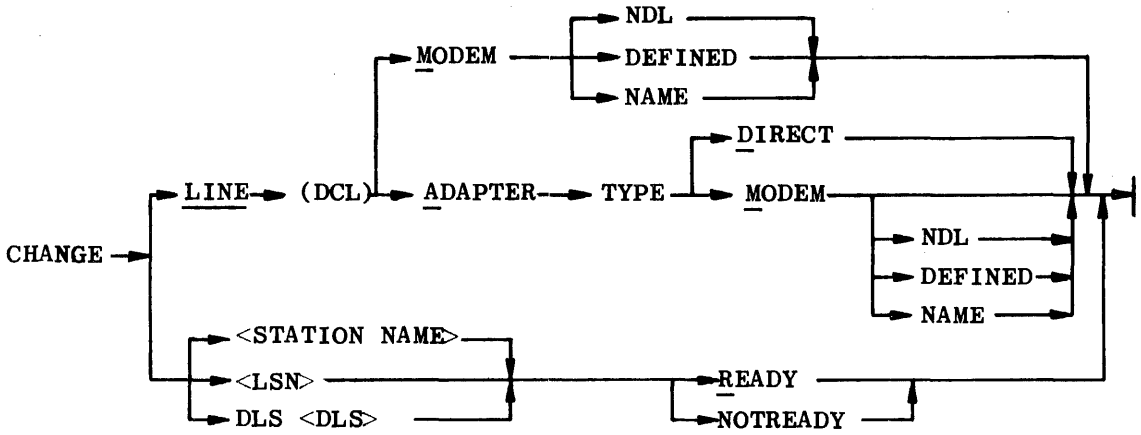
THE USER MAY NOW ACCESS THE "AI" REGISTER. THIS IS A VOLATILE REGISTER AND SHOULD NOT BE USED FOR DATA STORAGE. ITS MAIN PURPOSE IS TO ACCESS UNTRANSLATED BYTES RATHER THAN THE TRANSLATED BYTES IN THE CHARACTER REGISTER. THE FOLLOWING SYNTAX APPLIES:

<BYTE VARIABLE>::= <CHARACTER REGISTER> / <USER TALLY> /
<RETRY> / <BCC> / <CRC BYTE> / AI

DIAGNOSTICMCS (DIAGNOSTMCS)

D0564 DIAGNOSTMCS - CHANGE COMMAND - 11-18-73

THE CHANGE COMMAND IN DIAGNOSTICMCS HAS BEEN EXPANDED TO PROVIDE THE FACILITY TO CHANGE THE ADAPTER TYPE AND/OR MODEM ON A PARTICULAR LINE, AS WELL AS TO CHANGE LINE OR STATION STATUS FROM READY TO NOT READY OR VICE VERSA. THE FOLLOWING SYNTAX APPLIES:



WHERE: DCL = <DCP NUMBER>:<CLUSTER NUMBER>:<LINE NUMBER>

D0607 DIAGNOSTMCS - SOURCENDL VINTAGE - 12-28-73

THE DATACOM PREFIX AND SOURCENDL VINTAGE ARE INDICATED IN THE INITIAL "HELLO" MESSAGE SENT TO A STATION. THE SAME INFORMATION IS GIVEN IN THE HEADING OF MONITOR OUTPUT.

D0608 DIAGNOSTMCS - CLOSE PRINTFILE - 12-28-73

THE FACILITY TO CLOSE A PRINTFILE WHILE DIAGNOSTICMCS IS RUNNING IS GIVEN VIA THE CLOSE COMMAND. THE FOLLOWING SYNTAX APPLIES:

D0608 DIAGNOSTMCS - CLOSE PRINTFILE - 12-28-73

CLOSE <PRINTFILE ID>

WHERE

<PRINTFILE ID>::= LINE / PRINTER1 / PRINTER2 / PRINTER3 / PRINTER4 / PRINTER5

THIS, FOR EXAMPLE, ALLOWS A USER TO CLOSE A PRINTFILE CONTAINING MONITOR OR LINEANALYZER OUTPUT AND SUBSEQUENTLY PRINT THIS OUTPUT WITHOUT INTERRUPTING THE OPERATION OF DIAGNOSTICMCS.

D0665 DIAGNOSTMCS - CLUSTER ALL ORDERS TEST - 01-20-74

THE DATA COMMUNICATION PROCESSOR DIAGNOSTIC FACILITIES IN DIAGNOSTICMCS HAVE BEEN EXTENDED TO INCLUDE A CLUSTER ALL-ORDERS TEST. AT THE PRESENT, THE TEST IS LIMITED TO USE AT THE CPU SITE ONLY.. FUTURE CHANGES WILL ALLOW ACCESS FROM ANY STATION.

THE ALL-ORDERS TEST WILL PERFORM DIAGNOSTIC EVALUATION OF ALL AVAILABLE CLUSTERS IN THE DCP NETWORK IN THE SAME FASHION AS THE OFF-LINE CLUSTER TEST; THE FORCED IMAGE PATH LOGIC WILL BE VERIFIED FIRST, THEN THE TRANSLATION LOGIC. THE DIFFERENCE BEING THIS VERSION WILL BE ON-LINE AND CAN BE RUN IN CONJUNCTION WITH A FULLY OPERATIONAL DATACOM SYSTEM WITHOUT DISRUPTING THE OTHER STATIONS.

TO RUN, THE DCP(S) MUST BE INITIALIZED, AND THE MCS ENTERED AS A TASK VIA THE FOLLOWING CARDS:

```
<I>RUN SYSTEM/DIAGNOSTICMCS
<I>DATA
ALORDS <XXX>
QUIT
<I>END
```

THE <XXX> DESIGNATES A THREE DIGIT NUMBER REPRESENTING THE AMOUNT OF PASSES THE USER WOULD LIKE THE TEST TO RUN.

IN THE EVENT OF AN ERROR, THE USER WILL FIRST BE NOTIFIED AT THE SITE DISPLAY BY THE MESSAGE:

*** ALORDS ERR: CLUSTER # <N>

THE ERROR INFORMATION WILL BE RETAINED UNTIL THE END OF THE RUN, WHEN IT WILL BE LISTED ON THE LINE PRINTER. ALL NECESSARY ITEMS WILL BE SHOWN TO AID IN EVALUATING THE ERROR.

EXAMPLE:

*** DCP CLUSTER ALL-ORDERS ERROR ***

DCP/CLUSTER #00/00 TRANSLATION TEST *** PASS TALLY = 001
SOURCE WORD NUMBER: 096

	TYPE	BC/BI	SC/SA	CT/BT	CL	IR
SOURCE SENT:	02	40	01	00	00	00
RESULT EXPECTED:	22	4F	03	7F	00	N/A
RESULT RECEIVED:	22	4F	03	4C	80	N/A

****

WHERE PASS TALLY IS THE TEST PASS NUMBER THAT THE ERROR OCCURRED IN, SOURCE WORD NUMBER IS THE TEST CASE NUMBER OF THE SOURCE, AND THE VARIOUS FIELDS SHOWN ARE THOSE OF THE BUFFER ASSOCIATIVE REGISTER IN THE CLUSTER.

DUMPALL

D0675 DUMPALL - DUMPALL INITIAL RELEASE - 01-27-74

SYSTEM/DUMPALL IS A UTILITY FOR THE LISTING OF FILES, DUMPING OF TAPES AND COPYING OF FILES FROM ONE MEDIUM TO ANOTHER. TO OBTAIN A LIST OF THE INPUT SYNTAX:

```
RUN SYSTEM/DUMPALL("TEACH")
```

DUMPANALYZER

D0546 DUMPANALY - PRINTING RUN-TIME OPTIONS - 12-06-73

THIS PATCH IMPROVES THE PRINTING OF THE RUN-TIME OPTIONS IN TWO WAYS:

1. ALL OPTIONS ARE PRINTED, WHEREAS FORMERLY THEY WERE CHOPPED OFF AT THE END OF THE PRINT LINE.
2. UNKNOWN OPTIONS SET, I.E., WHOSE NUMBERS ARE GREATER THAN THE CURRENT MAXIMUM WILL BE IDENTIFIED BY NUMBER AND MARKED AS **UNKNOWN** IN THE SET OPTIONS SECTIONS.

D0576 DUMPANALY - CODEFILE NOW OPTIONAL - 12-06-73

THIS PATCH MAKES THE MCP CODEFILE OPTIONAL AND REMOVES THE RESIDENCY TEST. THE RESULT IS THAT OPERATOR INTERVENTION IN THE FORM OF LOADING THE MCP FROM TAPE OR MOUNTING A DISK PACK OR GIVING AN "OF" OR "IL" COMMAND IS NOW REQUIRED. THIS FEATURE IS MOST USEFUL FOR SYSTEMS WITH SMALL AMOUNTS OF DISK IN THAT THE MCP CAN BE LOADED WHEN NEEDED.

NOTE: AN IL MUST BE USED IF A DISK PACK CONTAINING THE CODEFILE IS BROUGHT ON LINE, SINCE THE DEFAULT KIND OF THE FILE IS DISK AND NOT PACK.

D0609 DUMPANALY - TOTAL MODS ON LINE COUNT - 12-28-73

THIS PATCH ALLOWS THE ANALYZER TO LIST THE TOTAL MEMORY MODS PRESENT.

D0610 DUMPANALY - PRINT TAPE SERIAL NUMBER - 12-28-73

DUMPANALYZER WILL NOW PRINT THE SERIAL NUMBER OF THE TAPE FROM WHICH THE DUMP IS TAKEN. THIS INFORMATION WILL FOLLOW THE LISTING OF THE TAPE DRIVE ON WHICH IT WAS MOUNTED.

D0627 DUMPANALY - MCS ANALYSIS - 01-09-73

THIS PATCH MODIFIES DUMPANALYZER TO CONFORM TO THE NEW FORMAT OF THE MCSTABLE IN THE MCP. SINCE THE MCSTABLE AND THE MCSNAMES ARRAYS ARE BOTH OVERLAYABLE, DUMPANALYZER WILL PRINT "MCS TABLE NOT PRESENT" OR "MCS NAME NOT AVAILABLE" WHEN APPROPRIATE TO INDICATE THE ABSENCE OF THE ARRAY(S) IN THE MEMORY DUMP.

ESPOL

D0590 ESPOL - CHECKPOINT DOLLAR OPTION - 10-28-73

A NEW ESPOL COMPILER $ CONTROL CARD OPTION, CHECKPOINT, HAS BEEN IMPLEMENTED. THE FORM IS $ CHECKPOINT N, WHERE N IS AN INTEGER. THIS NEW OPTION WILL CAUSE THE ESPOL COMPILER TO TAKE A CHECKPOINT EVERY N SECONDS (PLEASE REFER TO THE SYSTEM MISCELLANEA FOR AN EXPLANATION OF THE CHECKPOINT OPERATION). THE CHECKPOINT INFORMATION WILL BE KEPT ON DISK, AND EACH NEW CHECKPOINT WILL OVERRIDE THE PRIOR ONE (I.E., A CHECKPOINT(DISK,PURGE) IS USED). FOR EXAMPLE:

```
$SET LIST SINGLE CHECKPOINT 30 STACK
```

D0591 ESPOL - NEW ESPOL CONSTRUCT, CONVERT - 11-18-73

THIS CHANGE INTRODUCES THE NEW ESPOL (REAL) INTRINSIC, CONVERT (P, X). P IS A <DESTINATION> POINTER PART, AND X IS AN ARITHMETIC EXPRESSION. P MAY BE ANY OF THE FOLLOWING: Q, Q:S, Q:TOS, TOS:S AND TOS:TOS WHERE TOS IS TOP-OF-STACK AND Q AND S ARE POINTERS OR WORD IDENTIFIERS, OR POINTER OR WORD PROCEDURE IDENTIFIERS.

CONVERT (Q,X) PERFORMS AN INPUT CONVERT DESTRUCTIVE (ICVD) STARTING AT Q FOR X CHARACTERS.

CONVERT (Q:S,X) PERFORMS AN INPUT CONVERT UPDATE (ICVU) STARTING AT S FOR X CHARACTERS, AND STORES THE UPDATED POINTER INTO Q.

EXAMPLES:

```
Y:=CONVERT (Q,X);
Y:=CONVERT (Q:S,X);
```

FOR THE CASES INVOLVING TOS, THE USER SHOULD VERIFY THE CODE

EMITTED BY SETTING $ SET STACK CODE IN A SMALL SAMPLE PROGRAM, E.G., CONVERT (TOS:TOS,X) RETURNS THE UPDATED POINTER IN REGISTER A AND THE CONVERTED VALUE IN REGISTER B.

D0611 ESPOL - LINENUMBER TO ESPOL - 12-06-73

AN ESPOL USER MAY NOW FIND THE SEQUENCE NUMBER OF A CARD OR RECORD BEING COMPILED BY ASSIGNING THE KEYWORD LINENUMBER TO AN IDENTIFIER OF TYPE INTEGER, FOR EXAMPLE:

```
X:=LINENUMBER
```

PLACES THE CURRENT SEQUENCE NUMBER IN X.

D0666 ESPOL - $EXCLUDE TO ESPOL - 01-06-74

A NEW DOLLAR CARD OPTION, $EXCLUDE, HAS BEEN ADDED TO ESPOL. THIS COMPILE-TIME OPTION ALLOWS THE USER TO INHIBIT THE EMISSION OF CODE, PCW"S, AND SEGMENT DESCRIPTORS FOR WHOLE PROCEDURES. FOR EXAMPLE, $ EXCLUDE (PROC1,PROC2,PROC3) WOULD EXCLUDE THESE THREE PROCEDURES. IF AN EXCLUDED UNTYPED PROCEDURE IS CALLED, A COMPILE-TIME MESSAGE IS OUTPUT INDICATING THE FACT. IF A CALL IS MADE TO AN EXCLUDED TYPED PROCEDURE, AN ERROR MESSAGE IS PRODUCED. $ EXCLUDE CARDS ARE CUMULATIVE SO THAT THE ABOVE EXAMPLE COULD BE ACCOMPLISHED BY TWO CARDS:

```
$ EXCLUDE (PROC1)
$ EXCLUDE (PROC2,PROC3)
```

HOWEVER, $ EXCLUDE CARDS MUST BE COMPLETE ON ONE LINE, SO PROCEDURE NAMES MAY NOT RUN OVER MORE THAN ONE CARD. $ RESET EXCLUDE AND $ POP EXCLUDE ARE VALID. THESE WOULD CAUSE PREVIOUSLY EXCLUDED PROCEDURES NOT TO BE EXCLUDED. SIMILARLY, A $ CARD WITH NO SET SUCH AS $ MERGE WOULD CAUSE ALL PREVIOUSLY EXCLUDED PROCEDURES NOT TO BE EXCLUDED.

ESPOLINTRINSICS (ESPOLINTRN)

D0612 ESPOLINTRN - REMOVE AND CHANGE INTRINSICS - 12-28-73

THE 2.7 DIRECTORY REDESIGN WILL NOT ALLOW DIRECTORIES TO BE OPENED AS FILES. ANY EXISTING PROGRAMS WHICH OPEN DIRECTORIES SHOULD BE CHANGED BEFORE 2.7. THE FOLLOWING NEW INTRINSICS PROVIDE THE ABILITY TO CHANGE THE NAME OF AND REMOVE DIRECTORIES AND FILES WITHOUT OPENING THEM. THESE NEW INTRINSICS PLUS GETSTATUS WILL PROVIDE THE MECHANISMS NECESSARY FOR PROGRAM CONVERSIONS.

THE TWO NEW INTRINSICS "CHANGEFILE" AND "REMOVEFILE" HAVE BEEN IMPLEMENTED FOR ALGOL AND DCALGOL. THESE INTRINSICS WORK FOR BOTH DISK AND PACK FILES.

SYNTAX:

```
<REMOVEFILE STATEMENT>::= REMOVEFILE(<DIRECTORYELEMENT>)
<CHANGEFILE STATEMENT>::= CHANGEFILE(<DIRECTORYELEMENT>,
                                     <DIRECTORYELEMENT>)
<DIRECTORYELEMENT>::= <POINTER EXPRESSION> / <ARRAY ROW> / <STRING>
<STRING>::= "<FILETITLE>." / "<FILETITLE> ON <PACKNAME>."
```

SEMANTICS:

REMOVEFILE IS A BOOLEAN PROCEDURE WHICH RETURNS A VALUE OF TRUE IF AN ERROR OCCURRED. AN ERROR NUMBER FURTHER DEFINING THE FAILURE IS STORED IN THE FIELD [39:20]. THE FOLLOWING ERRORS ARE REPORTED FOR REMOVEFILE FAILURES:

1. 10 THE FILENAME WAS IN ERROR.
2. 30 THE FILENAME WAS NOT REMOVED.

IF A POINTER EXPRESSION IS USED AS A DIRECTORYELEMENT IT SHOULD POINT TO AN ARRAY WHICH CONTAINS THE FILETITLE TO BE REMOVED.

THE CHANGEFILE INTRINSIC IS ANALOGOUS TO THE REMOVEFILE INTRINSIC

D0612 ESPOLINTRN - REMOVE AND CHANGE INTRINSICS - 12-28-73

EXCEPT THAT A SECOND DIRECTORYELEMENT MUST BE SUPPLIED. THIS DESIGNATES THE TITLE TO WHICH THE FIRST TITLE WILL BE CHANGED. IF THE CHANGE IS ON A PACK THE SECOND TITLE MUST BE FOLLOWED BY "ON <PACKNAME>.". AN ERROR WILL BE RETURNED IF THE FIRST TITLE INCLUDES A PACKNAME. THE FOLLOWING ERRORS ARE REPORTED FOR CHANGEFILE FAILURES:

1. 10 THE FIRST FILENAME WAS IN ERROR.
2. 20 THE SECOND FILENAME WAS IN ERROR.
3. 30 THE FILENAME WAS NOT CHANGED.

EXAMPLE:

TO CHANGE A/B TO C/D AND THEN REMOVE C/D

```
BEGIN
     ARRAY A,B[0:44];
     BOOLEAN B;
     POINTER PA,PB;
     PA:=POINTER (A[0]);
     PB:=POINTER (B[0]);
     REPLACE PA BY 8"A/B.";
     REPLACE PB BY 8"C/D.";
     IF B:=CHANGEFILE(PA,PB) THEN ERROR;
     IF B:=REMOVEFILE(8"C/D.") THEN ERROR;
END.
```

D0680 ESPOLINTRN - G AND V FORMAT - 10-28-73

THIS PATCH ALLOWS THE VARIABLE FORMAT V TO BE EVALUATED BY EITHER A LEFT OR RIGHT JUSTIFIED LETTER, E.G., A "F" IN THE LIST (IN FORTRAN THIS IS "FSSSSS", WHERE S IS THE BLANK CHARACTER) EVALUATES A V10.2 TO AN F10.2, IN THE SAME MANNER AS DOES THE VALUE 198 (198 IS A RIGHT-JUSTIFIED "F" OVER A FIELD OF NULLS).

D0681 ESPOLINTRN - NEW U FORMAT - 10-28-73

THIS SYSTEM NOTE INTRODUCES THE NEW OUTPUT FORMAT PHRASE, U. THE U FORMAT PHRASE (U FOR UNIVERSAL) HAS THREE FORMS: U, UW, AND UW.D.

THE FORM U:

THIS FORM INSTRUCTS THE I/O SUBSYSTEM TO OUTPUT THE LIST ELEMENT (HEREAFTER, L.E.) IN WHATEVER FORMAT BEST REPRESENTS THE VALUE OF THE L.E. LOGICAL L.E."S ARE OUTPUT AS EITHER T OR F, AND OCCUPY ONE (1) CHARACTER POSITION IN THE OUTPUT RECORD. REAL, INTEGER, AND DOUBLE PRECISION L.E."S ARE OUTPUT IN A FORMAT THAT COMBINES MAXIMUM NUMERICAL SIGNIFICANCE WITH HIGH READIBILITY; THE CHARACTER POSITIONS OCCUPIED IN THE OUTPUT RECORD WILL VARY ACCORDING TO THE SIGN, MAGNITUDE, AND TYPE OF LIST ELEMENT (NOTE THAT A COMPLEX L.E. IS CONSIDERED TO BE TWO (2) CONSECUTIVE REAL L.E."S, AND THIS REQUIRES TWO (2) FORMAT PHRASES). STRING L.E."S ARE OUTPUT AS THE STRING ITSELF, E.G., "ABCDEFGHIJKLMN" WOULD BE OUTPUT AS ABCDEFGHIJKLMN, AND WOULD OCCUPY 14 CHARACTER POSITIONS (NOTE THAT STRINGS LESS THAN SEVEN CHARACTERS ARE TREATED AS REALS, AND STRINGS FROM SEVEN THROUGH 12 CHARACTERS ARE TREATED AS DOUBLES). IF THE NUMBER OF CHARACTERS REQUIRED TO EDIT THE L.E. IS GREATER THAN THE NUMBER LEFT IN THE CURRENT RECORD, THE CURRENT RECORD IS OUTPUT, AND THE L.E. IS OUTPUT ON THE NEXT RECORD, STARTING AT CHARACTER POSITION ONE FOR NON-PRINTER/REMOTE FILES, AND CHARACTER POSITION TWO FOR PRINTER OR REMOTE FILES; CHARACTER POSITION ONE WOULD IN THIS LATTER CASE CONTAIN A BLANK SPACE (NOTE THAT, WHILE THE U FORM EMPLOYS THAT FORMAT MOST SUITABLE FOR THE L.E., THE FORMAT IS CONSTRAINED SO AS NOT TO EXCEED THE NUMBER OF CHARACTERS IN THE RECORD).

THE FORM UW:

THIS FORM BEHAVES SIMILARLY TO THE FORM U, WITH THE ADDED CONSTRAINT THAT THE FORMAT EMPLOYED TO OUTPUT THE L.E. MUST

NOT EXCEED W CHARACTERS. FOR EXAMPLE, THE INTEGER VALUE 123456789, WHEN OUTPUT UNDER THE FORMAT U6, WOULD APPEAR AS 1.23E8. IF THE VALUE CANNOT BE EDITED WITHIN THE ALLOWABLE MAXIMUM FIELD WIDTH, A FIELD OF W ASTERISKS IS OUTPUT.

THE FORM UW.D

THIS FORM BEHAVES SIMILARLY TO THE FORM UW WITH THE ADDED CONSTRAINT THAT THE TOTAL FIELD WIDTH OCCUPIED BY THE EDITED L.E. MUST NOT BE LESS THAN D CHARACTERS. THUS, THE VALUE 12345678.9012, WHEN EDITED UNDER U10.5, U10.10, AND U10.15 WOULD PRODUCE, RESPECTIVELY, 12345678.9, 12345678.9, AND BBBBB12345678.9 (B IS THE BLANK CHARACTER); THE VALUE 123 WOULD PRODUCE RESPECTIVELY, BB123, BBBBBBB123, AND BBBBBBBBBBBB123. NOTE THAT W IS A CONSTRAINT ON THE WIDTH OF THE FORMAT, WHILE D IS A CONSTRAINT ON THE WIDTH OF THE OVERALL FIELD; THUS, W IS THE MAXIMUM NUMBER OF CHARACER POSITIONS AVAILABLE TO REPRESENT THE L.E., WHILE D IS THE MINIMUM NUMBER OF CHARACTER POSITIONS TO BE OCCUPIED IN THE RECORD. WHENEVER D IS GREATER THAN W, LEADING BLANKS WILL ALWAYS BE INSERTED; MOREOVER, WHENEVER D IS GREATER THAN OR EQUAL TO W, THE FIELD WIDTH IS A CONSTRAINT VALUE, D, INDEPENDENT OF THE TYPE OR MAGNITUDE OF THE L.E.

FORTRAN

D0694 FORTRAN - DOUBLE-TO-SINGLE OPTION - 01-06-74

THIS CHANGE IMPLEMENTS THE COMPILE-TIME DOLLAR OPTION, DBLTOSNGL, WHICH INTERPRETS EACH DOUBLE PRECISION CONSTRUCT AS THE ANALOGOUS SINGLE PRECISION CONSTRUCT.

D0706 FORTRAN - FORTRAN FILE ATTRIBUTES - 01-06-74

THIS ADDITION ALLOWS THE USER TO PROGRAMMATICALLY DEFINE AND INTERROGATE THE CHARACTERISTICS OF FILES. THIS IS PROVIDED VIA A NEW INTERFACE TO THE EXISTING FILE ATTRIBUTE SYSTEM AS DESCRIBED IN BURROUGHS B6700 INPUT/OUTPUT SUBSYSTEM INFORMATION MANUAL (DOCUMENT 5000185 OF MARCH 1974). ONLY THE INTERFACE, I.E., STATEMENT SYNTAX, WILL BE DELINEATED IN THIS NOTICE.

FILE CARDS

ALL PREVIOUS DOCUMENTATION CONCERNING FILE CARDS (APPENDIX A OF BURROUGHS B6700/B7700 FORTRAN REFERENCE MANUAL) IS STILL VALID. ADDITIONALLY, THE FIRST FILE INFORMATION CLAUSE MAY HAVE THE FORM OF AN UNSIGNED INTEGER FILE DESIGNATOR FROM ONE TO 99 FOLLOWED BY A LEFT PARENTHESIS, "(". IF THE FIRST CLAUSE IS OF THE ABOVE FORM THEN IT MUST BE FOLLOWED BY AT LEAST ONE ATTRIBUTE ASSIGNMENT CLAUSE OF THE FORM FA=V WHERE FA IS A VALID FILE ATTRIBUTE IDENTIFIER AS LISTED IN THE ABOVE REFERENCED "INPUT/OUTPUT SUBSYSTEM" MANUAL AND V IS AN EXPRESSION MNEMONIC WITH A VALID VALUE FOR THIS ATTRIBUTE WITH RESPECT TO TYPE AND RANGE. THE FIRST ATTRIBUTE ASSIGNMENT CLAUSE MAY BE FOLLOWED BY ADDITIONAL ATTRIBUTE ASSIGNMENTS SEPARATED BY COMMAS. THE LAST ATTRIBUTE CLAUSE MUST BE FOLLOWED BY A RIGHT PARENTHESIS, ")". FOR EXAMPLE:

```
FILE 6(TITLE="(USERCDE)MAS/FYLE",MAXRECSIZE=10,INTMODE=EBCDIC)
```

(NOTE THAT THIS NEW OPTIONAL FILE CARD FORMAT CLOSELY RESEMBLES THE FILE EQUATION CONSTRUCTS OF WORK FLOW MANAGEMENT.)

THOSE ATTRIBUTES WHOSE TYPE IS DENOTED AS "BOOLEAN" (I.E. "LOGICAL") CAN OPTIONALLY BE SET TRUE BY USE OF THE ATTRIBUTE IDENTIFIERALONE. FOR EXAMPLE:

```
FILE 7(TITLE="DISK/PACK",KIND=DISKPACK,CYLINDERMODE)
```

CYLINDERMODE, A BOOLEAN ATTRIBUTE, WILL BE SET TO TRUE.

"VALUE" INTRINSIC

AN IN-LINE VALUE INTRINSIC, IS AVAILABLE SO THAT THE USER MAY OBTAIN THE NUMERIC VALUES CORRESPONDING TO THE VARIOUS ATTRIBUTE VALUE MNEMONICS. VALUE HAS ONE PARAMETER WHICH IS A VALID ATTRIBUTE VALUE MNEMONIC AND RETURNS THE CORRESPONDING VALUE. IT MAY BE USED WHEREVER AN EXPRESSION MAY BE USED AND HAS THE SAME TYPE AS THE ASSOCIATED ATTRIBUTE. FOR EXAMPLE:

```
DEVICE=VALUE(DISK)

IF (DEVICE .EQ. VALUE(PRINTER)) GO TO 20
```

FILE ATTRIBUTE HANDLING

THIS IMPLEMENTATION REFLECTS OUR CURRENT EFFORT TO CONFORM TO THE AMERICAN NATIONAL STANDARD FORTRAN (X3.9-1966) SPECIFICATIONS. AT THIS WRITING MANY OF THE FILE ATTRIBUTE MNEMONICS AND THEIR CORRESPONDING VALUE MNEMONICS HAVE NOT BEEN COMPLETELY SPECIFIED; HENCE, THE MNEMONICS ALLOWED ARE ONLY THOSE WHICH CONFORM TO BURROUGHS INPUT/OUTPUT SUBSYSTEM SPECIFICATIONS WITH THE INTENTION OF ALLOWING APPROVED SYNONYMS AT A FUTURE RELEASE.

OPEN AND CHANGE STATEMENTS:

```
OPEN(N,FA1=E1,FA2=E2, ... ,FAN=EN)
CHANGE(N,FA1=E1,FA2=E2, ... ,FAN=EN)
```

WHERE N IS A FILE DESIGNATOR, FA IS A FILE ATTRIBUTE IDENTIFIER,

AND EACH E IS A VALID EXPRESSION WITH RESPECT TO THE TYPE AND RANGE OF THE FILE ATTRIBUTE IDENTIFIER. THESE STATEMENTS CAUSE THE LISTED ATTRIBUTES TO BE SET TO THE CORRESPONDING VALUES FOR THE INDICATED FILE. THE OPEN STATEMENT CAUSES AN EXPLICIT OPENING OF THE FILE; HENCE, THE FILE MUST BE CLOSED PRIOR TO THE EXECUTION OF THIS STATEMENT. INDEED, THE OPEN STATEMENT WITH FILE DESIGNATOR ONLY (I.E., OPEN(N)) WILL PERFORM AN EXPLICIT FILEOPEN WHEREAS THE CHANGE STATEMENT MUST INCLUDE AT LEAST ONE ATTRIBUTE ASSIGNMENT. FURTHER, THE EXPLICIT USE OF THE FILE ATTRIBUTE "OPEN" WITHIN THE OPEN STATEMENT CAUSES A WARNING DIAGNOSTIC AND THE SPECIFIED ACTION IS IGNORED. EXAMPLES ARE:

```
MRSZ=10
OPEN (1,KIND=DISK, MAXRECSIZE=MRSZ, AREAS=2)

M=VALUE(OUT)
CHANGE (1,MYUSE=M,AREACLASS(5+X)=Y)

FNUM=9;OPEN (FNUM)
```

INQUIRE STATEMENT:

FILE CHARACTERISTIC INFORMATION MAY BE OBTAINED VIA THE INQUIRE STATEMENT WHICH HAS THE FOLLOWING FORM:

```
INQUIRE (N,FA1=E1,FA2=E2, ... FAN=EN)
```

WHERE N IS A FILE DESIGNATOR, EACH FA IS A FILE ATTRIBUTE IDENTIFIER, AND EACH E IS A SIMPLE OR SUBSCRIPTED VARIABLE WHOSE TYPE CORRESPONDS TO THAT OF THE ATTRIBUTE. FOR EXAMPLE:

```
REAL KND,L(15)
LOGICAL OPE
INQUIRE(10,KIND=KND,OPEN=OPE,MAXRECSIZE=L(10),
      ROWADDRESS(0)=RADR)
```

AFTER EXECUTION OF THIS INQUIRE STATEMENT, KND WILL CONTAIN THE KIND OF FILE 10 (SOME KIND VALUES ARE SEVEN (7) OR VALUE(PRINTER), ONE (1) OR VALUE(DISK), ETC.), OPE WILL BE TRUE/FALSE AS FILE 10 WAS OPENED/CLOSED, AND L(10) WILL CONTAIN THE MAXIMUM RECORDSIZE IN CHARACTERS OR WORDS.

CLOSE STATEMENT:

THE CLOSE, PURGE, AND LOCK STATEMENTS AS PREVIOUSLY IMPLEMENTED ARE STILL VALID. ALTERNATIVELY, THE FOLLOWING FORMS MAY BE USED:

1. CLOSE (N)
2. CLOSE (N,DISP=DELETE)
3. CLOSE (N,DISP=KEEP)

WHERE N IS A FILE DESIGNATOR.

THE FIRST FORM IS EQUIVALENT TO THE ORIGINAL FORM OF THE CLOSE STATEMENT. THE SECOND FORM IS EQUIVALENT TO THE PURGE STATEMENT AND THE THIRD FORM IS EQUIVALENT TO THE LOCK STATEMENT. DISP STANDS FOR DISPOSITION.

GENERAL USER INFORMATION

"POINTER" FILE ATTRIBUTES, SOMETIMES CALLED "STRING" OR "ALPHA" ATTRIBUTES (E.G., TITLE, FORMMESSAGE, PACKNAME, FAMILY, UNITNAME, ETC.), ARE SET AND INTERROGATED USING DIMENSIONED VARIABLES. FOR EXAMPLE:

```
REAL A(2)/"MASTER/FILE."/
OPEN(7,TITLE=A)
```

(NOTE THE REQUIRED TERMINAL PERIOD)

FILE ATTRIBUTES THAT ALLOW PARAMETERS, SUCH AS ROWADDRESS AND AREACLASS, WILL ACCOMMODATE ANY ARITHMETIC EXPRESSION AS A PARAMETER VALUE. EXAMPLES:

```
ROWADDRESS(6)
AREACLASS(A(4,2)/Y+Z)
```

ALTHOUGH THE ATTRIBUTE SUBSYSTEM REQUIRES POINTER ATTRIBUTE INFORMATION TO BE IN EBCDIC MODE, SIX CHARACTERS PER WORD, THE COMPILER WILL NOT ATTEMPT TO PERFORM TRANSLATION OF STRING DATA CREATED WHILE EITHER BCL MODE WAS OPERATIVE (VIA $BCL) OR VIA $ CHARS OPTION. ALL ATTRIBUTE HANDLING FACILITIES ARE AVAILABLE FOR ANY LEVEL OF OPTIMIZATION AND FOR SEPARATE COMPILATIONS; THESE

CONSTRUCTS ARE ALSO AVAILABLE WITH THE MULTI-JOB COMPILATION OPTION WITH THE EXCEPTION THAT NO ATTRIBUTE HANDLING STATEMENT MAY REFERENCE THE DEFAULT OUTPUT PRINTER FILE.

INPUT/OUTPUT

D0565 MCP-I-O - FILE ATTRIBUTE REVISIONS - 11-18-73

SEVERAL FILE ATTRIBUTES HAVE BEEN MODIFIED, IMPLEMENTED, OR DELETED. EACH OF THE ATTRIBUTES IS LISTED WITH A CAPSULIZED DESCRIPTION OF HOW IT WORKS AND A BRIEF DISCUSSION OF THE CHANGES. THE CAPSULIZED DESCRIPTION CONTAINS THE FOLLOWING INFORMATION: NAME OF ATTRIBUTE AND ITS NUMBER IN PARENTHESES, THE ABILITY OF THE ATTRIBUTE TO BE ACCESSED (READ) OR SET (WRITTEN), DATA TYPE REQUIRED WHEN USING THE ATTRIBUTE (POINTER, INTEGER, BOOLEAN), CONDITIONS UNDER WHICH THE ATTRIBUTE MAY BE ACCESSED OR SET (CLOSED, OPEN, ASSIGNED, ANYTIME).

THE FOLLOWING ATTRIBUTES HAVE BEEN MODIFIED.

TITLE(0) GENERAL READ/WRITE POINTER ANYTIME/ANYTIME

THE TITLE ATTRIBUTE OF A LOGICAL FILE ASSIGNED TO A PERMANENT DISK FILE CAN BE CHANGED WHILE THE FILE IS OPENED; IN THIS CASE, BOTH THE TITLE OF THE PERMANENT FILE AS WELL AS THE LOGICAL FILE ARE CHANGED. THIS ATTRIBUTE CAN ALSO BE CHANGED WHEN THE FILE IS CLOSED EXCEPT WHEN IT IS STILL ASSIGNED TO A PERMANENT DISK FILE (E.G., IN ALGOL: DISKFILE.OPEN:=FALSE).

REEL(1) TAPE READ/WRITE INTEGER ANYTIME/CLOSED

THE REEL ATTRIBUTE, WHICH SPECIFIES THE REEL NUMBER (FILE SECTION NUMBER) OF A TAPE FILE, HAS AN ALLOWABLE RANGE BETWEEN THE DEFAULT VALUE OF ONE (1) AND 9999 INCLUSIVE. IT IS NOW SETTABLE ONLY WHEN THE FILE IS CLOSED.

DATE(2) DISK/TAPE READ ONLY INTEGER ASSIGNED

SINCE 2.0 (REFER TO TEMPORARY DOCUMENTATION 12-22-71), "TODAYSDATE" HAS ALWAYS BEEN USED AS THE CREATION DATE OF A FILE WHETHER OR NOT THE DATE ATTRIBUTE HAD BEEN SET BY THE PROGRAM. A FILE ATTRIBUTE ERROR IS NOW GIVEN WHENEVER THE

PROGRAM TRIES TO SET IT, THUS MAKING THE ATTRIBUTE READ ONLY.

SAVEFACTOR(5) TAPE/DISK READ/WRITE INTEGER ANYTIME/CLOSED

SINCE THE VALUE OF SAVEFACTOR IS USED ONLY AT THE TIME A FILE IS BEING OPENED, SAVEFACTOR MAY NOW ONLY BE SET WHILE THE FILE IS CLOSED.

DENSITY(6) TAPE ONLY READ/WRITE INTEGER ANYTIME/CLOSED

ALLOWS THE USER TO SPECIFY THE RECORDING DENSITY WHEN CREATING A MAGNETIC TAPE FILE. ACCESSIBLE AT ANY TIME. MAY BE SET ONLY WHEN FILE IS CLOSED. THE VALUES AND MNEMONICS OF THE DENSITY ATTRIBUTE ARE AS FOLLOWS:

0 HIGH (800BPI)
1 MEDIUM (556BPI)
2 LOW (200BPI)
3 SUPER (1600BPI)

MEDIUM IS NOT VALID FOR NINE TRACK TAPE (TAPE9). SUPER IS ONLY VALID FOR PHASE ENCODED TAPES (PETAPE). THERE ARE TWO EXCEPTIONS IN THE USE OF THE DENSITY ATTRIBUTE. WHEN CREATING A MULTI-FILE TAPE, THE DENSITY OF THE FIRST FILE IS USED FOR ALL SUBSEQUENT FILES. WHEN CREATING A MULTI-REEL FILE, THE DENSITY SETTING REMAINS CONSTANT FROM REEL TO REEL AS LONG AS IT IS VALID FOR THE TAPE UNIT.

LABEL EQUATABLE (DENSITY = <DENSITY MNEMONIC>). THE DEFAULT DENSITY VALUE IS THE DENSITY SETTING OF THE TAPE UNIT SELECTED FOR OUTPUT FILES AND THE DENSITY AT WHICH THE TAPE WAS WRITTEN FOR INPUT FILES. INITIALIZED BY FILE DECLARATION (ALGOL), (MAY BE SPECIFIED EITHER BY MNEMONIC OR VALUE).

SPEED(24) DISK READ/WRITE INTEGER ANYTIME/CLOSED

THE SPEED ATTRIBUTE SPECIFIES THE SPEED OF A HEAD-PER-TRACK DISK STORAGE UNIT. IT MAY NOW BE SET ONLY WHEN THE FILE IS CLOSED, SINCE ITS VALUE IS USED ONLY WHEN THE DISK FILE IS BEING OPENED. THIS ATTRIBUTE APPLIES TO HEAD-PER-TRACK DISK ONLY AND IS NOT VALID FOR PACK. NO SPEED TEST IS MADE IN

DISK-PACK USERDISK ALLOCATIONS SO FILES CREATED WITH THE SPEED ATTRIBUTE SET MAY BE COPIED TO ANY MODEL PACK.

CYLINDERMODE(41) PACK ONLY READ/WRITE BOOLEAN ANYTIME/CLOSED

WHEN THE ATTRIBUTE CYLINDERMODE IS TRUE, THE AREAS OF A NATIVE MODE DISK PACK FILE ARE ASSIGNED SO THAT THEY BEGIN UPON CYLINDER BOUNDARIES. THE DEFAULT VALUE IS FALSE.

LASTRECORD(49) DISK READ/WRITE INTEGER ANYTIME/CLOSED

INDICATES THE PHYSICAL RECORD NUMBER OF THE LAST RECORD IN A FILE. VALID ONLY FOR PERMANENT DISK FILES (THAT IS, FILES WHICH HAVE BEEN CLOSED, ENTERED IN THE DISK FILE DIRECTORY, AND THEN REOPENED). MAY BE SET AS WELL AS READ (ACCESSIBLE AT ANY TIME). IN ORDER TO SET THIS ATTRIBUTE THE FILE MUST BE CLOSED WITH UNIT RETAINED. THIS MAY BE ACCOMPLISHED VIA A REWIND STATEMENT OR BY SETTING THE OPEN ATTRIBUTE TO FALSE. IN ADDITION THE FILEKIND MUST BE EQUAL TO DATA, AND THE OPEN COUNT IN THE HEADER EQUAL TO ONE (1). DUPLICATED, CRUNCHED AND PROTECTED FILES ARE DISALLOWED.

FILEKIND(58) DISK READ ONLY INTEGER ANYTIME

TWO NEW FILEKINDS HAVE BEEN ADDED FOR THE USE OF DATA MANAGEMENT:

199 DBRESTARTSET
200 DBDATA

SERIALNO(62) PACK/TAPE READ ONLY REAL(WORD) ANYTIME

RETURNS THE SERIAL NUMBER OF THE LABELED TAPE OR PACK WHICH THE LOGICAL FILE IS ASSIGNED TO. THE SERIAL NUMBER OF A TAPE IS ESTABLISHED BY THE SN SYSTEM INPUT MESSAGE (SEE CONTROLLER D0562). THE IV, RC AND LB SYSTEM INPUT MESSAGES CAN BE USED TO ESTABLISH A SERIAL NUMBER FOR A PACK.

A TAPE SERIAL NUMBER IS AN ALPHANUMERIC STRING OF UP TO SIX CHARACTERS. A PACK SERIAL NUMBER CAN BE ONLY NUMERIC.

THE VALUE RETURNED BY THE SERIALNO ATTRIBUTE IS A WORD

CONTAINING THE SERIAL NUMBER IN EBCDIC CHARACTERS, LEFT-JUSTIFIED AND BLANK FILLED IF NECESSARY.

IF THE LOGICAL TAPE FILE IS UNLABELED, OR IF THE PHYSICAL TAPE HAS BEEN READ BEYOND THE LAST FILE ON THE TAPE, REFERENCING THE SERIALNO ATTRIBUTE CAUSES AN ATTRIBUTE ERROR.

WHEN AN UNLABELED TAPE FILE IS PURGED IT WILL BE GIVEN A SERIAL NUMBER OF "000000".

RECORDINERROR(84) DISK/TAPE READ ONLY INTEGER OPEN

THE RECORDINERROR ATTRIBUTE, WHICH INDICATES THE LOGICAL RECORD NUMBER OF THE BEGINNING RECORD IN THE TOP (I.E. CURRENT) BUFFER WHEN IT WAS LAST USED, IS NOW VALID FOR TAPE AS WELL AS DISK FILES.

IOINERROR(87) DISK/TAPE READ ONLY BOOLEAN OPEN

THE IOINERROR ATTRIBUTE, WHICH INDICATES WHETHER AN ERROR OCCURRED ON THE PHYSICAL I/O INVOLVING THE TOP BUFFER WHEN IT WAS LAST USED, IS NOW VALID FOR TAPE AS WELL AS DISK FILES.

DISPOSITION(111) REMOTE READ ONLY INTEGER OPEN

DATACOM STATION ATTRIBUTE. RETURNS THE DISPOSITION OF THE STATION IN THE REMOTE FILE AFTER MAKING SURE THE RSN IS VALID. ACCESSIBLE WHENEVER THE FILE IS OPEN.

THE FOLLOWING ATTRIBUTES HAVE BEEN DELETED.

UNITSLEFT(55)

THIS ATTRIBUTE HAS BEEN DELETED; A FILE ATTRIBUTE ERROR WILL NOW BE GIVEN WHENEVER UNITSLEFT IS REFERENCED. SINCE 2.0, AS STATED IN THE TEMPORARY DOCUMENTATION 12-22-71, THIS ATTRIBUTE HAS RETURNED A MEANINGLESS VALUE. UNITSLEFT WILL BE ELIMINATED FROM THE COMPILERS RESERVED WORD LIST ON A FUTURE RELEASE.

THE FOLLOWING ATTRIBUTES HAVE BEEN IMPLEMENTED.

CYCLE(3) DISK/TAPE READ/WRITE INTEGER ANYTIME/CLOSED

THE ATTRIBUTE CYCLE, IN CONJUNCTION WITH THE VERSION ATTRIBUTE, IS USED TO DENOTE THE DIFFERENT GENERATIONS OF A PERMANENT FILE. THE INITIAL AND DEFAULT CYCLE VALUE IS ONE (1). THE MAXIMUM VALUE FOR CYCLE IS 9999. THE VALUE OF THE CYCLE ATTRIBUTE CAN BE SET ONLY WHEN THE FILE IS CLOSED EXCEPT IN THE CASE OF THE CREATION OF A DISK FILE WHERE CYCLE CAN BE CHANGED ANYTIME BEFORE THE FILE IS ENTERED INTO THE DISK DIRECTORY.

SETTING THE CYCLE ATTRIBUTE TO ZERO (0) HAS THE SPECIAL EFFECT OF RESETTING BOTH THE CYCLE AND VERSION ATTRIBUTES TO THEIR DEFAULT VALUES AND MARKING THE LOGICAL FILE AS NOT REQUESTING SPECIFIC GENEALOGY CHECKING WHEN THE LOGICAL FILE IS ASSIGNED TO A PERMANENT FILE. IN THIS CASE, THE PERMANENT FILE WITH THE BEST GENEALOGY IS ASSIGNED TO THE LOGICAL FILE. BEST GENEALOGY IS DEFINED TO BE THE HIGHEST CYCLE AND THE HIGHEST VERSION OF THAT CYCLE.

IF CYCLE OR VERSION HAS BEEN SET TO A LEGITIMATE VALUE (THIS INCLUDES EXPLICITLY SETTING ONE OR BOTH THE ATTRIBUTES TO THEIR DEFAULT VALUES), ONLY A PERMANENT FILE WITH MATCHING GENEALOGY (ALONG WITH ALL THE OTHER PREREQUISITES) WILL BE ASSIGNED TO THE FILE. IF THE PROPER FILE CANNOT BE FOUND A "UNMATCHED GENEALOGY" NOTIFICATION IS GIVEN TO THE OPERATOR. THE OPERATOR CAN RESPOND BY MAKING THE FILE AVAILABLE, OR BY ENTERING A US OR DS SYSTEM INPUT MESSAGE.

PHYSICAL FILES WHICH DO NOT HAVE GENEALOGIES (CARD FILES FOR EXAMPLE) ARE ASSUMED TO HAVE DEFAULT GENEALOGY WHEN THE LOGICAL FILE DOES NOT CARE ABOUT A SPECIFIC GENEALOGY, AND ARE ASSUMED TO HAVE THE SPECIFIC GENEALOGY WHEN THE FILE DOES CARE.

VERSION(4) TAPE READ/WRITE INTEGER ANYTIME/CLOSED

THE ATTRIBUTE VERSION, IN CONJUNCTION WITH THE CYCLE ATTRIBUTE, IS USED TO DISTINGUISH ITERATIONS OF THE SAME GENERATION OF A PERMANENT FILE. ITS INITIAL AND DEFAULT

VALUE IS ZERO (0) AND ITS MAXIMUM VALUE IS 99.

TRANSLATE(91) GENERAL READ/WRITE INTEGER ANYTIME/CLOSED

THE ATTRIBUTE TRANSLATE INDICATES THE SCOPE OF TRANSLATION. THE VALUES, MNEMONICS AND MEANINGS OF THE TRANSLATE ATTRIBUTE ARE AS FOLLOWS:

0 DEFAULTRANS	THE DEFAULT: SOFTWARE TRANSLATION WILL TAKE PLACE ONLY IN THOSE CASES IN WHICH BEFORE 2.3 THE PROGRAM WOULD HAVE RECEIVED A FATAL INVALID TRANSLATION OPEN ERROR BECAUSE TRANSLATION WAS REQUIRED AND HARDWARE TRANSLATION WAS NOT PROVIDED.
1 FULLTRANS	SOFTWARE TRANSLATION WILL TAKE PLACE WHENEVER TRANSLATION IS REQUIRED AND HARDWARE TRANSLATION IS NOT PROVIDED.
2 SOFTONLY	SOFTWARE TRANSLATION WILL TAKE PLACE WHENEVER TRANSLATION IS REQUIRED.
3 FORCESOFT	SOFTWARE TRANSLATION USING TABLES THAT MAY BE PROVIDED BY THE PROGRAM WILL TAKE PLACE WHETHER OR NOT TRANSLATION IS REQUIRED AS LONG AS SOFTWARE TRANSLATION IS NOT SPECIFICALLY DISALLOWED (NEITHER INTMODE NOR EXTMODE IS EQUAL TO SINGLE, FILETYPE NOT EQUAL TO FOUR OR SIX, AND NOT DIRECT I/O FILE).
4 NOSOFT	WHENEVER TRANSLATION IS REQUIRED AND HARDWARE TRANSLATION IS NOT PROVIDED, GIVE FATAL INVALID TRANSLATION OPEN ERROR.
5 NOTRANS	WHENEVER TRANSLATION IS REQUIRED,

GIVE FATAL INVALID TRANSLATION OPEN ERROR.

TRANSLATING(92) GENERAL READONLY BOOLEAN ASSIGNED

THE ATTRIBUTE TRANSLATING RETURNS TRUE IF SOFTWARE TRANSLATION IS BEING PERFORMED ON THE RECORDS OF THE FILE. THE ATTRIBUTE IS SET OR RESET WHEN THE FILE ASSIGNMENT IS COMPLETE, TAKING INTO CONSIDERATION THE VALUES OF THE ATTRIBUTES INTMODE, EXTMODE, FILETYPE AND TRANSLATE.

INPUTTABLE(93) GENERAL WRITE ONLY TRANSLATETABLE ANYTIME

SETTING THE ATTRIBUTE INPUTTABLE ALLOWS THE PROGRAM TO MODIFY THE INPUT SOFTWARE TRANSLATION OF THE LOGICAL I/O SUBSYSTEM. THE INPUTTABLE IS ONLY USED WHEN THE TRANSLATING ATTRIBUTE IS TRUE, I.E., TRANSLATION IS BEING PERFORMED ON THE RECORDS OF THE FILE; OTHERWISE, THE TRANSLATE TABLE (IF ONE IS ASSIGNED) IS IGNORED. A TRANSLATE TABLE ASSIGNED TO INPUTTABLE WHICH IS DECLARED IN A PROGRAM MUST EITHER BE THE FIRST TABLE IN A LIST OF TABLES, OR THE ONLY TABLE IN THAT DECLARATION.

EACH TIME THE LOGICAL FILE IS CLOSED, INPUTTABLE REVERTS TO ITS DEFAULT SYSTEM VALUE.

AN EXAMPLE OF THE ALGOL SYNTAX FOR ASSIGNING A TRANSLATE TABLE TO INPUTTABLE IS AS FOLLOWS:

"FILEID.INPUTTABLE:= ASCIITOBCD;"

OUTPUTTABLE(94) GENERAL WRITE ONLY TRANSLATTABLE ANYTIME

SETTING THE ATTRIBUTE OUTPUTTABLE ALLOWS THE PROGRAM TO MODIFY OUTPUT SOFTWARE TRANSLATION OF THE LOGICAL I/O SUBSYSTEM. THE OUTPUTTABLE IS ONLY USED WHEN THE TRANSLATING ATTRIBUTE IS TRUE, I.E., TRANSLATION IS BEING PERFORMED ON THE RECORDS OF THE FILE; OTHERWISE, THE TRANSLATE TABLE (IF ONE WAS ASSIGNED) IS IGNORED. A TRANSLATE TABLE ASSIGNED TO OUTPUTTABLE WHICH IS DECLARED IN A PROGRAM MUST EITHER BE THE FIRST TABLE IN A LIST OF TABLES, OR THE ONLY TABLE IN THAT DECLARATION.

EACH TIME THE LOGICAL FILE IS CLOSED, OUTPUTTABLE REVERTS TO ITS DEFAULT SYSTEM VALUE.

AS EXAMPLE OF THE ALGOL SYNTAX FOR ASSIGNING A TRANSLATE TABLE TO OUTPUTTABLE IS AS FOLLOWS:

"FILEID.OUTPUTTABLE:= BCDTOASCII;"

D0580 MCP-I-O - FILE GENEALOGY - 11-18-73

THE FOLLOWING CHANGES HAVE BEEN MADE WHICH AFFECT FILE GENEALOGY:

1. IN AN "UNMATCHED GENEALOGY" MESSAGE, THE CYCLE AND VERSION VALUES DESIRED BY THE LOGICAL FILE WILL ALSO BE DISPLAYED. ITS FORM WILL BE "<CYCLE NUMBER>:<VERSION NUMBER>".
2. IF A CYCLE VALUE OF ZERO AND VERSION VALUE OF ZERO ARE ENTERED WITH A "US" INPUT MESSAGE, THE PERMANENT FILE WITH THE BEST GENEALOGY WILL BE USED IN THE FILE ASSIGNEMENT.
3. IMPLEMENTED WITH A RELATED PATCH TO THE MCP, THE REPLY TO AN "OL" OR "PER MT" <UNIT NUMBER> WILL NOW SHOW THE CYCLE AND VERSION VALUES OF THE MAGNETIC TAPES MOUNTED, IF THEIR VALUES ARE OTHER THAN THE DEFAULT VALUES. THIS INFORMATION FOLLOWS THE REEL NUMBER WITH THE CYCLE NUMBER FIRST FOLLOWED BY A COLON, THEN THE VERSION NUMBER, E.G., MT 17*7[000017] #1 2:15 RMC.

D0594 MCP-I-O - FULL SOFTWARE TRANSLATION - 12-06-73

IN GENERAL, THE TRANSLATION OF THE RECORDS IN A PHYSICAL FILE IS REQUIRED WHENEVER THE MODE OF THE RECORDS IN THE PHYSICAL FILE (EXTMODE) IS DIFFERENT FROM THE MODE OF THE LOGICAL FILE (INTMODE). SOFTWARE TRANSLATION IS RESTRICTED TO THE CASES WHERE THE DATA IN THE FILE IS STORED IN CHARACTER (HEX, BCL, EBCDIC OR ASCII) MODE; BINARY DATA CANNOT BE TRANSLATED AND MUST REMAIN UNALTERED IN ITS

48 BIT PATTERN. THAT IS TO SAY, LOGICAL FILES WHERE EITHER INTMODE OR EXTMODE EQUALS SINGLE (0) OR WHERE, IN THE CASE OF CARD FILES, EXTMODE EQUALS BINARY (6) ARE NEVER TRANSLATED. THE FORTRAN LINKED RECORD FORMAT (FILETYPE EQUAL SIX) WHICH IMPLIES THAT THE DATA IS BINARY, AND THE GENERALIZATION OF THE COBOL VARIABLE LENGTH RECORD FORMAT (FILETYPE EQUAL FOUR) WHICH ALLOWS THE MIXING OF MODES IN THE RECORDS, DEFY THE POSSIBILITY OF TRANSLATION. DIRECT I/O FILES DO NOT RECEIVE SOFTWARE TRANSLATION. IT SHOULD BE NOTED THAT THE ORIENTATION OF THE DATA TRANSFER (WORD OR CHARACTER) DEFINED BY THE UNITS AND INTMODE ATTRIBUTES IS NOT RELEVANT AT THIS POINT TO THE DISCUSSION OF THE TRANSLATION OF THE DATA ITSELF.

THE LOGICAL I/O SUBSYSTEM TAKES ADVANTAGE OF HARDWARE TRANSLATION IN THE CASES WHERE IT IS PROVIDED (FOR EXAMPLE, CARD AND PRINTER FILES) WHENEVER INTMODE EQUALS EBCDIC AND EXTMODE EQUALS BCL.

SEVEN TRACK TAPE FILES HAVE TWO CASES OF TRANSLATION OR TRANSFORMATION WHICH NO OTHER PHYSICAL UNIT TYPE HAS. NON-STANDARD (ALPHA, EVEN) PARITY SEVEN TRACK TAPES ALWAYS HAVE EXTMODE EQUAL TO BCL AND THE HARDWARE TRANSLATORS ALLOW THE INTMODE TO BE EITHER BCL OR EBCDIC. STANDARD PARITY TAPES COMING FROM THE B3500 REQUIRE BCL TO BCL OR EBCDIC TO EBCDIC TRANSFORMATION THROUGH THE HARDWARE. ALL OTHER COMBINATIONS ARE HANDLED IN 6-BIT BINARY. WHEN IT MATTERS TO THE HARDWARE, AS IT DOES IN THE NON-STANDARD PARITY CASE, ALL INTMODES OTHER THAN BCL ARE ASSUMED TO HAVE AN 8-BIT FRAMESIZE.

PAPER TAPE FILES HAD A SPECIAL USE FOR THE COMBINATION INTMODE EQUAL EBCDIC, EXTMODE EQUAL ASCII THAT WAS IMPLEMENTED BEFORE THE SYSTEM HANDLED 8-BIT ASCII. NOW, UNLESS TRANSLATE EQUALS DEFAULTTRANS, ASCII WILL BE TRANSFERRED IN 8-BIT BINARY EXACTLY AS EBCDIC IS HANDLED.

BEFORE SOFTWARE TRANSLATION WAS IMPLEMENTED (2.3) THE LOGICAL I/O SUBSYSTEM WAS INCONSISTENT IN WHETHER IT WOULD DISALLOW LOGICAL FILES TO BE OPENED THAT REQUIRED TRANSLATION WHEN IN FACT NO TRANSLATION WAS PROVIDED. BECAUSE OF THIS THE DEFAULT INVOCATION OF SOFTWARE TRANSLATION IS CONFINED TO THOSE CASES WHERE PROGRAMS

WOULD HAVE BEEN TERMINATED WITH AN INVALID TRANSLATION OPEN ERROR PRIOR TO 2.3.

THE TRANSLATE ATTRIBUTE GIVES THE PROGRAM CONTROL OVER THE INVOCATION OF TRANSLATION, ALL THE WAY FROM DISALLOWING ALL TRANSLATION HARDWARE OR SOFTWARE TO REQUIRING SOFTWARE TRANSLATION (POSSIBLY TO SCRAMBLE THE DATA) WHEN NO TRANSLATION IS REQUIRED.

WHETHER OR NOT TRANSLATION IS INVOKED FOR THE FILE, THE ATTRIBUTES MAXRECSIZE, BLOCKSIZE AND AREASIZE ARE IN TERMS OF THE LOGICAL UNITS OF THE FILE AS DEFINED BY THE INTMODE AND UNITS ATTRIBUTES. BOTH CHARACTER AND WORD ORIENTED DATA TRANSFER ARE ALLOWED WHILE THE RECORDS ARE BEING TRANSLATED. THE PROGRAMMER SHOULD BE AWARE THAT WORD ORIENTED FILES (WHERE MACRECSIZE, ETC., ARE DESCRIBED IN TERMS OF WORDS) WHERE THE FRAME SIZES DIFFER BETWEEN INTERNAL AND EXTERNAL MODE REQUIRE EITHER CONTRACTION OR EXPANSION OF THE RECORDS SUCH THAT THE LOGICAL AND PHYSICAL RECORD AND BLOCK SIZES ARE NOT THE SAME. AN EXAMPLE OF THIS IS THE DIFFERENCE BETWEEN AN EBCDIC AND A BCL CARD FILE WHICH HAVE RECORD SIZES OF 14 AND 10 WORDS RESPECTIVELY.

THE FILETYPE ATTRIBUTE ALLOWS THE MOST GRACEFUL SOLUTION TO THE PROBLEM OF EXPANSION AND CONTRACTION OF DATA DURING SOFTWARE TRANSLATION. SETTING FILETYPE TO EIGHT (8) CAUSES THE LOGICAL I/O SUBSYSTEM TO ALLOCATE ITS BUFFERS IN TERMS OF THE BLOCKING OF THE PERMANENT FILE AND MODIFIES THE ATTRIBUTES MAXRECSIZE AND BLOCKSIZE SO THAT THE LOGICAL FILE MAPS NEATLY INTO THE PERMANENT FILE.

D0595 MCP-I-O - SYSTEM USE OF KINDLISTS - 12-06-73

WITH THIS PATCH, A KINDLIST IS GENERATED WHENEVER THE FILE ATTRIBUTE KIND IS SET BY THE PROGRAM. SINCE THIS IS SIMILAR TO WHAT CONTROLCARD GENERATES FOR THE KIND SETTING IN A LABEL EQUATION CARD, THIS PATCH WILL ALLOW A PROGRAM TO OVERRIDE ITS LABEL-EQUATED KIND VALUE WHICH WAS PREVIOUSLY IMPOSSIBLE. COBOL PROGRAMS WHICH SELECT BACKUP TAPE FOR THEIR PRINTER FILE WILL NOW ALSO WORK.

D0617 MCP-I-O - DIRECT I-O BUFFER ATTRIBUTE - 12-28-73

I/O CANCEL IS NOW AVAILABLE TO DIRECT I/O TO ABORT AN I/O WHICH HAS BEEN REQUESTED. SETTING THE BUFFER ATTRIBUTE IOCANCEL TO TRUE ON A DIRECT BUFFER WHICH HAS A REQUESTED I/O WILL CAUSE THE UNIT ASSOCIATED WITH THAT BUFFER TO BE CLEARED. ALL PRESENT AND FUTURE I/OS TO THAT UNIT WILL RETURN A CANCELED RESULT. IOCANCEL IS NOT VALID FOR DISK, DISK PACK OR REMOTE DEVICES.

D0695 MCP-I-O - GLOBAL FILES FOR IPC PROGRAMS - 02-03-74

WORK FLOW LANGUAGE GLOBAL FILE EQUATION WAS DISALLOWED FOR IPC CAPABLE PROGRAMS BECAUSE OF SOME INHERENT PROBLEMS WITHIN THE MCP WHICH COULD CAUSE SYSTEM FAILURE WHEN THEY WERE USED. THESE PROBLEMS HAVE BEEN CORRECTED, AND WORK FLOW LANGUAGE GLOBAL FILES ARE NOW ALLOWED FOR ALL FILES EXCEPT DIRECT FIB"S.

LISTDIRECTORY (LISTDIR)

D0597 LISTDIR - BAD PARAMETER DETECTION - 12-06-73

INPUT PARAMETERS (TASK VALUE) GREATER THAN TWO ARE NOW SYNTAX CHECKED AND ZERO IS USED. BEFORE THIS, THE PARAMETER WAS CHECKED AS ODD, EVEN OR ZERO. HENCE ANYTHING WOULD WORK. SEE LISTDIRECTORY D0344 IN D NOTE DOCUMENTATION 10/73.

D0632 LISTDIR - FLAG CONFLICTS IN REPORT #4 - 01-09-73

MISLEADING RESULTS IN THE REPORT DEALING WITH AREAS WHICH CAN BE MADE AVAILABLE BY REMOVING ONE FILE ARE NOW FLAGGED. THESE PROBLEMS ARE THE RESULT OF FILE CONFLICTS.

D0633 LISTDIR - SYSTEM IDENTITY IN HEADING - 01-09-73

THE SYSTEM NUMBER OF THE COMPUTER ON WHICH LISTDIRECTORY IS RUNNING IS NOW INCLUDED IN THE PAGE HEADINGS.

LISTFILES

D0712 LISTFILES - IDENTIFY SYSTEM FILES - 02-10-74

THIS PATCH RESULTS IN ALL SYSTEM FILES HAVING TWO OR MORE NAMES (A/B) BEING IDENTIFIED BY PRECEDING THE OUTERMOST DIRECTORY NAME BY AN ASTERISK (*). ALL SINGLY NAMED SYSTEM FILES WILL NOT BE SO IDENTIFIED.

LOADER

D0545 LOADER - AUTO BAD FILE REMOVAL-LOADER - 10-28-73

WHEN ATTEMPTING TO RECOVER FROM A DAMAGED DIRECTORY USING THE LOADER, EACH TIME THE LOADER ENCOUNTERED A BAD FILE IT WOULD ASK FOR AN "OK". NOW WHEN IT ENCOUNTERS A BAD FILE THE MESSAGE "ENTER OK OR GO" WILL BE DISPLAYED. WHEN "GO" IS ENTERED, ALL BAD FILES WILL BE REMOVED WITHOUT FURTHER OPERATOR INTERVENTION.

D0566 LOADER - RECOGNITION OF TRAIN PRINTERS - 11-18-73

CHANGES HAVE BEEN MADE TO THE LOADER TO REFLECT A NEW TYPE OF PRINTER, THE TRAIN PRINTER. THE LOADER LOADS THE EBCDIC 64 TRANSLATE TABLE TO THE FIRST PRINTER IT FINDS IF IT IS A TRAIN PRINTER. THE TRANSLATE TABLE MAY ALSO BE LOADED BY A PARAMETER CARD REQUEST, E.G.,

```
LOAD TRANSLATETABLE LP N EBCDIC 64;
```

WHERE N IS THE UNIT NUMBER OF THE LINE PRINTER.

THIS PATCH ALSO SETS THE TESTCPU OPTION BY DEFAULT.

D0614 LOADER - LOADER ONLY LOADS MCPCODE FILE - 12-28-73

THIS PATCH ALLOWS THE LOADER TO ONLY LOAD FILES WITH A FILEKIND OF MCPCODEFILE (DCP LOADS ARE NOT AFFECTED).

D0676 LOADER - NEW MCP OPTIONS IN LOADER - 01-27-74

THIS PATCH ALLOWS THE SETTING OR RESETTING OF THE OPTIONS FULLTRANSLATION AND BACKUPBYJOBNR BY THE LOADER (MCP OPTIONS).

MAKEUSER

D0567 MAKEUSER - INPUT-OUTPUT CAPABILITY - 11-18-73

THE INPUT/OUTPUT CAPABILITY OF MAKEUSER HAS BEEN BOTH IMPROVED AND EXPANDED. IT IS NOW POSSIBLE TO RUN MAKEUSER FROM A REMOTE DEVICE. A USER MAY SET A NEW DOLLAR OPTION, $HARDCOPY, WHICH WILL WRITE A COPY OF ALL INPUT FROM AND OUTPUT TO THE TERMINAL TO A PRINTER FILE.

THE COPY FUNCTION HAS BEEN EXPANDED TO ALLOW A USER TO COPY THE USERDATAFILE TO PUNCHED CARDS AND/OR A DISK FILE AS WELL AS TO THE PRINTER. (COPY PUNCH DISK <FILENAME> PRINTER).

A NEW DOLLAR OPTION, $INPUT <FILENAME>, WILL CAUSE MAKEUSER TO READ FROM THE SPECIFIED INPUT FILE UNTIL AN END STATEMENT OR END-OF-FILE. END WOULD TERMINATE MAKEUSER; END-OF-FILE CAUSES READING TO REVERT TO THE CARD FILE. IN THE EXAMPLE

```
CREATE
$ INPUT X
```

MAKEUSER WOULD CREATE A NEW USERDATAFILE FROM INPUT SAVED IN FILE X, WHICH MAY HAVE BEEN WRITTEN BY MAKEUSER WITH A "COPY DISK X" COMMAND.

A NEW DOLLAR OPTION, $ADD, CAUSES MAKEUSER TO INFER A "+" BEFORE ANY USER SEGMENT (UNLESS A "-" IS EXPLICITLY PRESENT). THUS, IN UPDATE MODE THE DEFAULT OPERATION BECOMES "ADD NEW USER" RATHER THAN "MODIFY USER".

ANOTHER NEW DOLLAR OPTION, $SAVE <FILENAME>, MAY BE SET TO CAUSE ALL MAKEUSER INPUT (EXCEPT DISKFILE FROM $INPUT, OR RESPONSE TO BREAK-ON-OUTPUT INTERACTION) TO BE WRITTEN TO THE NAMED FILE. $ RESET SAVE (WITHOUT FILENAME) TURNS THE OPTION OFF. IF THE NAMED FILE IS ALREADY PRESENT, NEW RECORDS ARE ADDED TO THE END OF IT; OTHERWISE, THE FILE WILL BE CREATED. THIS FEATURE MAY BE USED TO

D0567 MAKEUSER - INPUT-OUTPUT CAPABILITY - 11-18-73

PRESERVE THE INPUT FROM A TERMINAL, FOR DIAGNOSTIC OR AUDITING PURPOSES.

D0568 MAKEUSER - $SYNTAX - 11-18-73

$SYNTAX MAY BE SET TO CAUSE MAKEUSER TO FULLY PROCESS ITS INPUT WITHOUT ACTUALLY MODIFYING OR CREATING THE *SYSTEM/USERDATAFILE.

D0569 MAKEUSER - INCREMENT LOCATOR TYPE - 11-18-73

IT IS NOW POSSIBLE TO USE MAKEUSER TO INCREMENT OR DECREMENT THE VALUE OF A FIELD, REAL, WORD, OR SUBSCRIPTED ARRAY ITEM.

EXAMPLES:

```
T1R=4      %ASSIGNMENT: VALUE IS FOUR
T1R+7      %INCREMENT: VALUE IS NOW 11
T1R-6      %DECREMENT: VALUE IS NOW FIVE
```

D0570 MAKEUSER - RECALL; PURGE DISPOSTION - 11-18-73

TWO NEW FUNCTIONS HAVE BEEN IMPLEMENTED. RECALL <FILENAME> CHANGES THE SPECIFIED FILE TO *SYSTEM/USERDATAFILE AND THE MCP IS MADE COGNIZANT OF THE CHANGE. DISPOSITION OF THE EXISTING USERDATAFILE MAY BE SPECIFIED AS WITH CREATE OR WITH COPY NEW. PURGE MAY BE SPECIFIED AS A DISPOSITION FOR ANY OF THE FOLLOWING FUNCTIONS : CREATE, RECALL, AND COPY NEW.

D0713 MAKEUSER - CONTROL INPUT WITHOUT "$" - 02-20-74

THE SPECIFICATIONS FOR SOME CONTROL INPUT TO MAKEUSER HAVE BEEN CHANGED: THE "CREATE" MODE AND THE "COPY" FUNCTION ARE NOW INVOKED AS COMMANDS IN THE MAKEUSER LANGUAGE RATHER THAN ON DOLLAR-SIGN

CARDS. THIS CHANGE RESULTS IN A CLEANER LANGUAGE DESIGN AND A CORRESPONDINGLY CLEANER PROGRAM: SUBSTANTIVE OPERATIONS ARE NOW INVOKED THROUGH THE NORMAL LANGUAGE, WHILE DOLLAR CARDS CONTROL AUXILIARY FUNCTIONS SUCH AS LISTINGS AND ERROR RECOVERY.

THE NET CHANGES VISIBLE TO PRE-2.6 MAKEUSER USERS ARE:

1. THE DOLLAR SIGN MUST BE OMITTED BEFORE "CREATE" AND "COPY" SPECIFICATIONS, AND OTHER DOLLAR CONTROLS (SUCH AS "LIST") MAY NOT APPEAR ON THE SAME CARD. THE SUBORDINATE SYNTAX WITHIN THESE SPECIFICATIONS REMAINS (FOR EXAMPLE, "COPY PRINTER NEW").

2. THE LIST OF RESERVED KEYWORDS NOW INCLUDES "CREATE" AND "COPY", PLUS "RECALL" AND "ACCESS" (FOR FUNCTIONS NOT PRESENT IN 2.5). NONE OF THESE WORDS MAY BE ASSIGNED AS A LOCATOR IDENTIFIER OR USED AS A DEFINE IDENTIFIER.

MCP

D0543 MCP - NEW SYSTEM MESSAGES - 10-28-73

THE FOLLOWING NEW SYSTEM MESSAGES HAVE BEEN ADDED. THOSE MARKED WITH AN ASTERISK ARE PRECEDED BY <MIX INDEX>.

OUTPUT MESSAGES

CHECKPOINT RESTART MESSAGES

THE FOLLOWING MESSAGES MAY APPEAR WHEN A RESTART IS AUTOMATICALLY ATTEMPTED BY THE SYSTEM AFTER A HALT/LOAD OR BY A RERUN STATEMENT. PLEASE REFER TO THE SYSTEM MISCELLANEA FOR FURTHER INFORMATION ON CHECKPOINT RESTART.

*RESTART PENDING

PLEASE REFER TO THE DESCRIPTION IN THE RSVP SECTION.

*MISSING CHECKPOINT FILE
*IO ERROR DURING RESTART
*USERCODE NO LONGER VALID
*OPERATOR DSED RESTART
*OPERATOR QTED RESTART
*MISSING CODE FILE
*NOT ABLE TO RESTART
*INVALID JOB FILE
*ERR COPYING JOB FILE
*RESTART AS CP/NNNN
*MISSING JOB FILE
*FILE POSITIONING ERROR
*WRONG JOB FILE
*WRONG CODE FILE
*BAD CHECKPOINT FILE

*BAD STACK NUMBER
*WRONG MCP

THE FOLLOWING MESSAGES MAY APPEAR WHEN A JOB IS CHECKPOINTED PROGRAMMATICALLY:

*CHECKPOINT #NNN
*INVALID AREA IN STACK
*SYSTEM ERROR
*BAD IPC ENVIRONMENT
*NO USER DISK FOR CP FILE
*IO ERROR DURING CHECKPOINT
*<INTEGER> ROWS IN CP FILE > 1024
*DIRECT FILE NOT ALLOWED
*TOO MANY TEMPORARY DISK FILES
*PAPER TAPE FILE NOT ALLOWED
*DUPLICATED FILE NOT ALLOWED
*CON FILE NOT ALLOWED
*CARD PUNCH FILE NOT ALLOWED
*OPEN REVERSED TAPE FILE NOT ALLOWED
*DISKHEADER IN STACK
*DMS AREA IN STACK
*DIRECT ARRAY IN STACK
*DIRECT DOPE VECTOR IN STACK
*SUBSPACE IN STACK
*STACKMARK
*SORT AREA IN STACK
*REMOTE FILE NOT ALLOWED
*ILLEGAL CONSTRUCT
*BDBASE ILLEGAL
*TEMP FILE ON NAMED PACK

*CLUSTER #N FAILED

THIS MESSAGE IS DISPLAYED FOR EACH CLUSTER THAT IS NOT ON LINE AND READY AT THE TIME OF DCP INITIALIZATION.

*DIRECTORY CAPACITY EXCEEDED

THIS MESSAGE OCCURS WHEN ALL THE ROWS OF A DIRECTORY HAVE BEEN USED AND AN ATTEMPT IS MADE TO ENTER A FILENAME. THE PROGRAM WILL BE DS-ED.

PK <NNN> <FILE ID1> [NOT] CHANGED TO <FILE ID2>

AN ATTEMPT TO CHANGE THE NAME OF THE GIVEN IC DISK-PACK FILE HAS FAILED (NOT) OR SUCCEEDED AS INDICATED.

PK <NNN> <FILE ID> DIR READ ERR, RD=<XXXXX>, SEG=<NNNNNNN>
HDR WRITE

AN I/O ERROR HAS OCCURRED WHILE WORKING WITH THE DIRECTORY (DIR) OR A HEADER (HDR). <XXXXX> IS THE RESULT DESCRIPTOR AND <NNNNNNN> IS THE SEGMENT ADDRESS. ERRORS DURING A WRITE (WRITE) MAY HAVE DAMAGED THE PACK.

PK <NNN> <FILE ID> FAULT

A FAULT HAS OCCURRED DURING PROCESSING BY AN IC DISK-PACK DIRECTORY MANAGEMENT ROUTINE. A PROGRAMDUMP IS PRODUCED IF THE MCP WAS COMPILED WITH $DIAGNOSTICS. THE ROUTINE ATTEMPTS TO ABORT THE PROCESS AND EXIT.

PK <NNN> <FILE ID> [NOT] REMOVED

AN ATTEMPT TO REMOVE OR PURGE AN IC DISK PACK HAS FAILED (NOT) OR SUCCEEDED AS INDICATED.

*<FILE ID> SECURITY CHANGED

THIS MESSAGE OCCURS WHEN A SECURITY CHANGE HAS OCCURRED.

*<FILE ID> SECURITY NOT CHANGED (NOT FOUND)

THIS MESSAGE OCCURS WHEN AN ATTEMPTED SECURITY CHANGE HAS NOT OCCURRED.

TRAIN PRINTER ERROR MESSAGES

THE FOLLOWING MESSAGES CONCERNING TRAIN PRINTERS HAVE BEEN IMPLEMENTED TO FACILITATE ERROR RECOVERY:

LP <N> INVALID RESULT DESCRIPTOR <RD>

LP <N> UNKNOWN TRAIN ID <RD>

LP <N> UNABLE TO LOAD TRAIN TABLE <RD>

THIS MESSAGE OCCURS AFTER THE MCP HAS MADE FOUR ATTEMPTS TO LOAD THE TRAIN TABLE.

*UNEXPECTED HEYU

THE OCCURRENCE OF THIS MESSAGE MEANS THAT A BAD HEYU INTERRUPT WAS RECEIVED FROM THE DCP (SEE P2728).

RSVP MESSAGES

FIX IT OR USE ?? <MIX INDEX> DS

THIS MESSAGE OCCURS WHEN A "PK <NNN> (MIX#) WRITE LOCKOUT" MESSAGE HAS BEEN DISPLAYED BUT THE OPERATOR HAS NOT YET REPLIED. THE MESSAGE WILL APPEAR AFTER 15 SECONDS AND EVERY 60 SECONDS THEREAFTER UNTIL THE OPERATOR WRITE-ENABLES THE PACK OR DS-ES THE PROGRAM.

PK <NNN> <FILE ID> DUP FILE

AN ATTEMPT WAS MADE TO ENTER THE GIVEN IC DISK-PACK FILE INTO THE DIRECTORY BUT IT CONFLICTED WITH A FILE ALREADY THERE. THE OPERATOR MAY

1. REPLY DS.
2. REMOVE OR CHANGE THE CONFLICTING FILE AND THEN REPLY OK.
3. SET OPTION 5 (AUTORM) AND REPLY OK.

PK <NNN> (MIX#) NOT READY

THE OCCURRENCE OF THIS MESSAGE MEANS THAT THE PACK IS NOT READY. THE OPERATOR MAY

1. REPLY OK, THE OPERATION WILL BE RETRIED.
2. REPLY DS, THE OPERATION IS TERMINATED AND POSTED

WITH A RESULT DESCRIPTOR OF NINE (9).

PK <NNN> <FILE ID> REQ <MM> SECTORS

AN ATTEMPT WAS MADE TO ENTER A NEW IC DISK-PACK FILE IN THE DIRECTORY BUT THERE WAS NO SPACE ON THE PACK TO EXTEND THE DIRECTORY. THE OPERATOR MAY

1. REPLY DS.
2. MAKE SURE THE PACK IS WRITE ENABLED, ENTER A "MODE PK <NNN> OUT" COMMAND AND THEN OK.
3. REMOVE A FILE ON THE PACK AND THEN REPLY OK.

PK <NNN> <FILE ID> WAITING FOR HEADER SLOT

AN ATTEMPT WAS MADE TO OPEN THE GIVEN IC DISK-PACK FILE BUT THERE WAS NOT ENOUGH ROOM IN CORE TO STORE THE HEADER. THE OPERATOR MAY

1. REPLY DS.
2. WAIT FOR SOME FILE TO BE CLOSED AND THEN REPLY OK.

PK <NNN> (MIX#) WRITE LOCKOUT

THE OCCURRENCE OF THIS MESSAGE MEANS THAT A WRITE LOCKOUT ERROR HAS OCCURRED. THE OPERATOR MAY

1. REPLY OK, THE OPERATION WILL BE RETRIED.
2. REPLY DS, THE OPERATION IS TERMINATED AND POSTED WITH ITS ERROR.

PK <NNN> (MIX#) WRONG PACK: <MESSAGE>

WHERE

```
<MESSAGE>::= I/O ERR, RD=<XXXXX> /
             PK HAS NO LABEL /
             [SERIAL] PACKNAME
```

1. I/O ERR, RD=XXXXX
2. PK HAS NO LABEL

3. [SERIAL] PACKNAME

THIS MESSAGE OCCURS WHEN IT APPEARS TO THE SYSTEM THAT THE PACK MAY HAVE BEEN DISMOUNTED AND ANOTHER PACK MOUNTED IN ITS PLACE AND AN ATTEMPT TO VERIFY THAT THE CORRECT PACK IS STILL MOUNTED (BY READING THE VOLUME LABEL) HAS FAILED. THE OPERATOR MAY

1. USE THE OL PKNN COMMAND TO DETERMINE WHICH PACK IS REQUIRED, REMOUNT IT, AND REPLY OK.
2. REPLY DS. THIS CAUSES THE PACK TO BE MARKED AS "BLASTED" AND ALL SUBSEQUENT I/O OPERATIONS WILL BE CANCELLED WITH A RESULT DESCRIPTOR OF ONE (1). ONCE A PACK IS BLASTED THE DRIVE MAY NOT BE USED UNTIL THE UNIT IS SUCCESSFULLY CLOSED AND READIED.
3. TURN THE PACK OFF, REPLY OK, AND THEN DS THE SUBSEQUENT NOT READY. THIS AVOIDS BLASTING THE PACK.

*REQ *PK NNN PACKNAME.

THE OCCURRENCE OF THIS MESSAGE MEANS THAT A WRITE-ENABLED DISK PACK (THE ASTERISK IS USED TO INDICATE WRITE-ENABLED) IS REQUIRED FOR OUTPUT. THE OPERATOR MAY

1. MOUNT A WRITE-ENABLED PACK WITH THE CORRECT PACKNAME.
2. CREATE A PACK WITH THE CORRECT NAME WITH THE RC OR LB INPUT MESSAGE.
3. DS THE PROGRAM.

*REQUIRES <FILE ID> BACKUP(<BACKUP LIST>) <OUTPUT DEVICE>

WHERE

<BACKUP LIST>::= PK / PK <PACKNAME> / DK / MT / MT7
MT9 / MTPE
<OUTPUT DEVICE>::= LP / CP

THIS MESSAGE OCCURS WHEN A PROGRAM HAS ASKED FOR A PARTICULAR TYPE OF BACKUP MEDIUM AND THAT TYPE IS NOT AVAILABLE. THE

OPERATOR MAY

1. USE THE OU MESSAGE.
2. MAKE THE SPECIFIED PERIPHERAL READY.
3. USE THE SB COMMAND TO EQUATE A BACKUP MEDIUM THAT IS PRESENT AND AVAILBLE. DOING AN SB CAUSES THE WAITING PROCESS TO SEARCH AGAIN FOR AN AVAILABLE BACKUP MEDIUM.
4. DS THE PROGRAM.

*RESTART PENDING

THIS MESSAGE OCCURS AFTER A HALT/LOAD WHEN THE SYSTEM ATTEMPTS TO RESTART A JOB CONTAINING CHECKPOINTS. IF OK IS ENTERED THE JOB WILL BE RESTARTED AT THE LAST CHECKPOINT. DS WILL CAUSE THE JOB NOT TO BE RESTARTED. QT WILL SAVE THE CHECKPOINT FILE FOR LATER RESTART IF IT WAS CHECKPOINT WITH PURGE.

*UNMATCHED GENEALOGY <CYCLE NUMBER>:<VERSION NUMBER>

THIS MESSAGE IS DISPLAYED WHEN THE CYCLE AND VERSION VALUES DESIRED BY THE LOGICAL FILE ARE INCORRECT. THE OPERATOR MAY

1. USE THE IL MESSAGE.
2. USE THE US MESSAGE.
3. USE THE DS MESSAGE.

RESPONSES TO OPERATOR INPUT

OU RESPONSES

SEVERAL OU INPUT MESSAGES ARE INAPPROPRIATE GIVEN THE NATURE OF THE MEDIA. A LIST OF THE POSSIBLE RESPONSES BY THE SYSTEM TO SUCH INPUT IS AS FOLLOWS:

IS DIRECT FILE: CANNOT BACKUP

A DIRECT FILE ASKING FOR A LINE PRINTER OR CARD PUNCH CANNOT GO TO BACKUP UNDER ANY CIRCUMSTANCES.

THAT PK IS NOT PRESENT

AN OUPK FAILED BECAUSE THE PACKNAME SPECIFIED IN THE ATTRIBUTE LIST OF THE FILE COULD NOT BE FOUND ON THE SYSTEM. THIS MESSAGE WILL ALSO BE GENERATED WHEN A PACK OF THAT NAME IS PRESENT BUT IS INAPPROPRIATE BECAUSE IT IS AN INTERCHANGE, A WRITE-LOCKED OUT, OR A CONTINUATION PACK.

REQUIRES PK WITH CORRECT NAME

AN ATTEMPT WAS MADE TO OUPK <NNN> FOR A FILE WITH THE PACKNAME ATTRIBUTE SET, AND PK <NNN> DID NOT HAVE THE CORRECT NAME.

PK <PACKNAME> IS NOT PRESENT

AN ATTEMPT WAS MADE TO OUPK WHEN NO <PACKNAME> WAS PROVIDED BY THE PROGRAMMER AND NO DESIGNATED SYSTEM RESOURCE PACK WAS PRESENT.

REQD PKUNIT NOT MOUNTED-BP

THE OPERATOR DIRECTED OUPK <NNN> TO A PACK THAT WAS NOT A MOUNTED NATIVE MODE BASE PACK. BACKUP PACK MUST BE:

1. MOUNTED
2. WRITE-ENABLED READY
3. NATIVE MODE
4. BASE PACK

NEED AN OUTPUT TAPE FOR OUMT

OUMT FAILED BECAUSE NO TAPE WAS IN THE PROPER STATE.

D0547 MCP - LOGGING CHANGES - 10-28-73

UNITS ATTRIBUTE IN LOG ENTRY FOR CLOSE

THE SYSTEM LOG ENTRY FOR FILE CLOSE HAS BEEN CHANGED TO CONTAIN THE VALUE FOR UNITS WHICH WILL GIVE MORE MEANING TO THE MAXRECSIZE, MINRECSIZE, AND BLOCKSIZE VALUES THAT WERE ALREADY CONTAINED IN THE FILECLOSE ENTRY.

WORD 14 [3:4] OF LOG CLOSE ENTRY CONTAINS THE UNITS APPLICABLE THE MAXRECSIZE, MINRECSIZE AND BLOCKSIZE. THE VARIABLE INFORMATION FOR CLOSE RECORDS NOW BEGINS AT WORD 15 OF THE ENTRY. BITS [47:44] OF WORD 14 ARE AVAILABLE FOR FURTHER EXPANSION.

BASE PACK SERIAL NUMBER IN LOG ENTRY FOR CLOSE

LOG ENTRIES FOR CLOSE PREVIOUSLY CONTAINED ZERO FOR THE SERIAL NUMBER FOR PACKS. THIS CHANGE CAUSES THE BASE PACK SERIAL NUMBER TO BE LOGGED.

TABLE 3-2, PAGE 3-16 IN THE WORK FLOW USERS GUIDE SHOULD BE UPDATED SO THAT WORD 13 HAS THE FOLLOWING ENTRY FORMAT:

TAPE SERIAL NUMBER FOR A TAPE FILE AND BASE PACK SERIAL NUMBER FOR A PACK FILE (SIX EBCDIC CHARACTERS); ZERO FOR OTHER UNITS.

PACK BIT ADDED TO COMPARE ERROR RECORDS

THIS PATCH ADDS A BIT TO THE LOG RECORDS PRODUCED FOR COPY/COMPARE ERRORS. BIT 41 OF WORD FOUR WILL BE ONE IF THE SOURCE WAS A DISK PACK AND BIT 41 OF WORD SIX WILL BE ONE IF THE DESTINATION WAS A DISK PACK.

ADDITION OF WORDS 20 AND 21 TO EOT

VARIABLE LENGTH DATA IN THE EOT LOG ENTRY NOW BEGINS IN WORD 22. WORD 20 IS NOW THE AVERAGE WORKINGSET (CODE). WORD 21 IS THE AVERAGE WORKINGSET (DATA).

D0548 MCP - LOGPOSITIONING OPTION - 10-28-73

TAPEPARITYRETRY NOW HAS AN OPTION WHICH ENABLES LOGGING OF THE RESULT DESCRIPTORS INVOLVED IN WRITEPARITY TAPE POSITIONING. THE OPTION IS ACTIVATED BY THE COMPILE TIME OPTION LOGPOSITIONING. ANY RETRY WHICH IS ABORTED WILL LOG THE RESULT WHICH PRECEDED THE ABORT.

D0549 MCP - USERDATA CHANGES - 10-28-73

SYNTAXONLY MODE

IF BIT [7:1] OF THE USERDATA "ACTION" (FIRST) PARAMETER IS SET, USERDATA WILL GO THROUGH ALL THE MOTIONS WITHOUT ACTUALLY MODIFYING THE *SYSTEM/USERDATAFILE.

NAME-FILENAME FORMAT

THE FETCH AND EXAMINE OR EXAMINE FUNCTION OF USERDATA NOW RETURNS ITEMS OF TYPE "NAME" OR "FILENAME" IN EXTERNAL (DISPLAY) FORM, UNLESS THE "STANDARD" BIT IS SET IN THE "ACTION" PARAMETER. THE UDVLENF VALUE IS THE LENGTH IN CHARACTERS.

ERROR CODE

THE ERROR CODE FOR CASES WHEN "FIRSTPW" IS NOT CORRECT HAS BEEN CHANGED FROM 18 (OLD PASSWORD NEEDED) TO 19 (OLD PASSWORD INCORRECT), FOR CONSISTENCY WITH DOCUMENTATION.

INCREMENT FUNCTION

DATA ITEMS OF THE TYPE WORD, REAL OR FIELD MAY NOW BE INCREMENTED AS WELL AS ASSIGNED A VALUE, WHEN AN ENTRY IS MODIFIED THROUGH USERDATA FUNCTION SEVEN OR DIRECTLY THROUGH USERDATAREBUILD. THE UDLTYPER OR THE APPROPRIATE LOCATOR WORD MUST BE CHANGED TO FIVE TO CAUSE INCREMENTING.

D0549 MCP - USERDATA CHANGES - 10-28-73

FIELD OVERFLOW

FIELD AND BIT ASSIGNMENTS ARE NOW TESTED FOR OVERFLOW; IF THE ASSIGNED (OR INCREMENTED) VALUE IS TOO LARGE FOR THE RECORD AN ALARM ERROR IS RETURNED.

LIST EXAMINATION

THE EXAMINATION OF A USERDATA ENTRY FOR MEMBERSHIP OF A SPECIFIED ITEM IN A LIST (WORD LIST, NAME LIST OR FILE LIST) MAY NOW BE PERFORMED WITH FUNCTION ONE (EXAMINE SUPPLIED ENTRY) AS WELL AS FUNCTION TWO (FETCH & EXAMINE). THE TARGET ITEM MUST BE SUPPLIED IN THE FOURTH PARAMETER (OUTSTUFF), NOT THE FIFTH (INSTUFF).

NOTE THAT SOME OF THESE CHANGES ALTER THE EXTERNAL SPECIFICATIONS OF PREVIOUSLY RELEASED BUT UNDOCUMENTED FEATURES.

D0550 MCP - DISK-PACK ERROR RECOVERY - 10-28-73

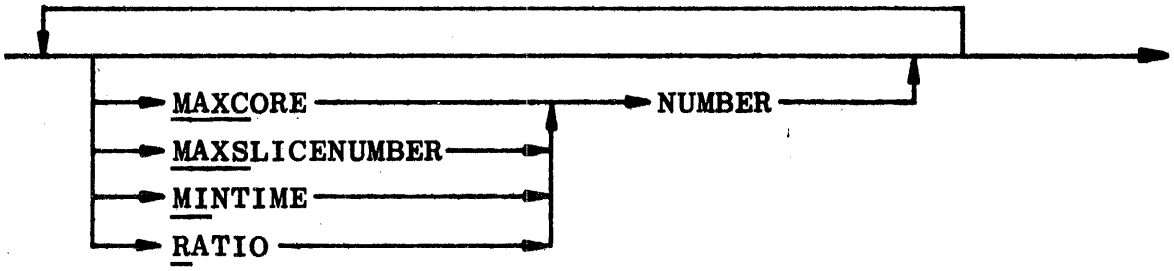
THE DOCUMENTATION FOR DISK-PACK ERROR RECOVERY WILL BE FOUND IN THE SYSTEM MISCELLANEA.

D0555 MCP - SWAPPER - 10-28-73

PARAMETERS

SWAPPER NOW HAS AN ADDITIONAL EXECUTION PARAMETER CALLED MAXCORE. THIS VARIABLE IS THE MAXIMUM SUBSPACE SIZE (IN WORDS) WHICH WILL BE ALLOWED FOR A JOB. THE VALUE OF THIS PARAMETER WILL BE TAKEN FROM WORD SEVEN OF RECORD ZERO IN SYSTEM/SWAPDISK, AND MUST BE A MULTIPLE OF THE CORE CHUNK SIZE (990 WORDS). IF THE VALUE IN WORD SEVEN IS ZERO, THE MAXCORE WILL BE SET TO THE MINIMUM OF THE SUBSPACE SIZE OR 130,680 WORDS.

THE PARAMETERS SWAPPER USES DURING SWAPPING (MAXCORE, MINTIMESLICE, MAXSLICENUMBER AND RATIO) MAY NOW BE SET DYNAMICALLY WHILE SWAPPER IS EXECUTING. THESE PARAMETERS ARE SET VIA AN ACCEPT INPUT TO SWAPPER. THE SYNTAX FOR THIS INPUT IS:



THE UNDERLINED CHARACTERS REPRESENT MINIMUM RECOGNIZED ABBREVIATIONS FOR THE PARAMETER.

<NUMBER>: IS ANY INTEGER OR DECIMAL NUMBER OF 12 OR FEWER DIGITS. DECIMAL NUMBERS MUST HAVE ONE DIGIT (ZERO IS PERMITTED) BEFORE THE DECIMAL POINT AS THE SCANNING IGNORES SPECIAL CHARACTERS SEPARATING IDENTIFIERS AND NUMBERS.

MAXCORE IS EXPRESSED IN CORE CHUNKS, AND MAY NOT BE LESS THAN TWO OR GREATER THAN THE NUMBER OF CORE SLOTS IN THE TOTAL SUBSPACE OR 132, WHICHEVER IS LESS.

SWAPPER WILL DISPLAY MESSAGES FOR ANY ERRORS ENCOUNTERED AND WILL DISPLAY THE CURRENT SETTING FOR THESE PARAMETERS AFTER PROCESSING THE INPUT. NOTE: AN EMPTY ACCEPT MESSAGE IS SUFFICIENT TO INTERROGATE THE CURRENT SETTINGS.

FILE ADDRESSING LIMITATION

FILES FOR SWAP JOBS MAY NOT BE GREATER THAN 65535 WORDS FROM THE BASE OF THE SUBSPACE. IF THIS CONDITION IS NOT MET, THE PROGRAM WILL BE DS-ED FOR MEMORY EXCEEDED.

D0556 MCP - CHECKPOINT - RESTART - 10-28-73

THE DOCUMENTATION FOR CHECKPOINT RESTART WILL BE FOUND IN THE SYSTEM MISCELLANEA.

D0557 MCP - CPUTEST CHANGES - 10-28-73

DISPLAY UPDATE TEST CASES HAVE BEEN ADDED TO CPUTEST.

THE HARDWARE J-COUNT LOGIC OF THE ARITHMETIC OPERATORS (ADD, SUBTRACT, AND MULTIPLY) IS NOW CHECKED WITH INDIVIDUAL TEST CASES. EACH TEST CASE IS DESIGNED TO CHECK A CERTAIN PATH OF J-COUNT LOGIC.

LARGE SECTIONS OF THE CPUTEST HAVE BEEN RESEQUENCED.

D0571 MCP - TASK PRIORITY - 11-18-73

SINCE JOBS, TASKS AND QUEUES ALL HAVE THE POTENTIAL FOR PRESCRIBING PRIORITY, SOME QUESTION MAY EXIST AS TO THE ACTUAL PRIORITY A GIVEN TASK WILL RUN WITH. THIS NOTE DESCRIBES THE RULES.

1. IF A TASK SPECIFIES A PRIORITY, IT WILL RUN WITH THAT PRIORITY, ASSUMING IT IS WITHIN THE QUEUE LIMIT.
2. IF A TASK DOES NOT SPECIFY A PRIORITY, IT WILL RUN WITH THE PRIORITY OF THE JOB. THE PRIORITY OF THE JOB IS DEFINED AS FOLLOWS:

   A. IF THE JOB SPECIFIES A PRIORITY, THAT WILL BE ITS PRIORITY.

   B. IF A JOB DOES NOT BEGIN WITH A JOB CARD, THERE IS NO WAY FOR THE USER TO SPECIFY ITS PRIORITY. HOWEVER, THE WORK FLOW LANGUAGE COMPILER SUPPLIES A JOB PRIORITY IF ANY TASK HAS SPECIFIED A PRIORITY, NAMELY THE MAXIMUM PRIORITY SPECIFIED FOR ANY TASK. THE PURPOSE OF THIS IS TO PREVENT THE JOB FROM ENTERING A QUEUE WHOSE PRIORITY LIMIT IS TOO

LOW FOR ONE OF ITS TASKS.

C. IF CASES (A) AND (B) DO NOT APPLY, THE JOB WILL BE RUN WITH THE QUEUE DEFAULT PRIORITY, IF SUPPLIED, OTHERWISE WITH THE SYSTEM DEFAULT OF 50.

EXAMPLES:

ASSUME ALL JOBS BELOW ARE RUN IN A QUEUE WITH A DEFAULT PRIORITY OF 49. THEN

1. ?JOB X; PRIORITY = 51
   BEGIN
   COMPILE A/B ALGOL;
   ALGOL PRIORITY = 56;
   DATA
   .
   .
   .
   ?END JOB

   THE COMPILE WILL RUN AT PRIORITY 56 AND THE GO AT 51.

2. ?JOB X;
   BEGIN
   COMPILE A/B ALGOL;
   ALGOL PRIORITY = 56;
   DATA
   .
   .
   .
   ?END JOB

   THE COMPILE WILL RUN AT 56 AND THE GO AT 49, THE QUEUE PRIORITY.

3. ?COMPILE A/B ALGOL
   ?ALGOL PRIORITY = 56
   ?DATA
   .

.
.
?END

BOTH COMPILE AND GO WILL RUN AT 56 BECAUSE USING RULE 2A, THE PRIORITY OF THE JOB WILL BE SET TO 56.

D0572 MCP - ENHANCED DATA MANAGEMENT - 11-18-73

THE DOCUMENTATION FOR THE ENHANCED BURROUGHS DATA MANAGEMENT SYSTEM MAY BE FOUND IN THE FOLLOWING DOCUMENTS:

B6700/B7700 DATA AND STRUCTURE DEFINITION LANGUAGE (DASDL) REFERENCE MANUAL (5000821)
B6700/B7700 DMSII HOST LANGUAGE INTERFACE (ALGOL AND COBOL) TO BURROUGHS DATA MANAGEMENT SYSTEM (5000839)

D0574 MCP - TRAIN PRINTERS - 11-18-73

THIS PATCH ALLOWS RECOGNITION OF TRAIN PRINTERS BY THE MCP. THESE PRINTERS DIFFER FROM OTHER PRINTERS IN THAT THE CHARACTER SET (ON A TRAIN MODULE) IS AN EASILY REMOVABLE DEVICE. THIS MODIFIABILITY, HOWEVER, REQUIRES THAT A SOFTWARE TRANSLATE TABLE BE LOADED WHICH IDENTIFIES THE CHARACTER SET ON A PARTICULAR TRAIN. ONCE A TRAIN MODULE IS MOUNTED THE MCP WILL ENSURE THAT ITS RESPECTIVE TRANSLATE TABLE IS KEPT LOADED. IN ADDITION TO THE MCP MECHANISM THAT LOADS THE TRANSLATE TABLE, THE OPERATOR CAN ENTER ON THE CONSOLE, LT LP N EBCDIC 64, WHERE N IS THE UNIT NUMBER OF THE LINE PRINTER. THE SYSTEM WILL RESPOND WITH "LP N EBCDIC 64 TRANSLATE TABLE LOADED". A DISPLAY OF THE TRAIN MODULE INSTALLED ON A TRAIN PRINTER CAN BE OBTAINED WITH THE "P-" OR "P LP" KEYBOARD INPUT MESSAGE.

THE FOLLOWING MESSAGES HAVE BEEN IMPLEMENTED TO FACILITATE TRAIN PRINTER ERROR RECOVERY:

LP <N> INVALID RESULT DESCRIPTOR <RD>

D0574 MCP - TRAIN PRINTERS - 11-18-73

LP <N> UNKNOWN TRAIN ID <RD>
LP <N> UNABLE TO LOAD TRAIN TABLE <RD>

SYSTEM/LOADER ALSO HAS THE MECHANISM TO LOAD AN EBCDIC TRANSLATE TABLE TO A TRAIN PRINTER. REFER TO LOADER SYSTEM NOTE D0556.

D0577 MCP - BACKUP TO PACK - 11-18-73

THIS PATCH PROVIDES AUTOBACKUP OF PRINTER AND PUNCH FOR DISK PACK IN ADDITION TO CURRENTLY IMPLEMENTED DISK. THIS FEATURE IS MORE FULLY DESCRIBED IN THE SYSTEM MISCELLANEA.

THIS PATCH ALSO IMPLEMENTS RUN-TIME MCP OPTION "BACKUPBYJUBNR".

PLEASE REFER TO THE CHAPTER ON DISK PACKS IN THE SYSTEM MISCELLANEA.

D0581 MCP - TASK ATTRIBUTES - 11-18-73

NEW TASK ATTRIBUTES

NUMBER	NAME	TYPE	CLASS
48	RESTARTED	BOOLEAN	2

FOR A JOB, THIS ATTRIBUTE WILL BE UNCONDITIONALLY SET AFTER A HALT/LOAD. FOR A TASK, THIS ATTRIBUTE WILL BE SET WHEN THE TASK IS INITIATED VIA A RERUN STATEMENT AFTER CHECKPOINT.

44	DESTNAME	STRING	2

THIS IS A STRING ATTRIBUTE WHICH MAY BE SET TO ANY STATION NAME DEFINED IN THE NETWORK DEFINITION LANGUAGE (NDL) OR THE WORD "SITE." WHEN SET TO A STATION NAME, THE NDL DESCRIPTION WILL BE INTERROGATED AND THE DESTINATION LSN AND CONTROLLING MCS NUMBER WILL BE SET IN THE PATH CONTROL WORD. (DESTMCSF, DESTUNITF) IF SET TO "SITE.", THEN THE DESTINATION CONTROL PORTION OF THE PATH CONTROL WORD WILL BE SET TO ZERO.

THIS ATTRIBUTE MAY ALSO BE READ, IN WHICH CASE IT WILL RETURN THE STATION NAME ASSOCIATED WITH THE DESTINATION UNIT. IF INTERROGATED AND A REMOTE DESTINATION HAS NOT BEEN SPECIFIED, IT WILL RETURN THE STRING "SITE."

46 DESTSTATION INTEGER 2

THIS IS AN INTEGER VALUED ATTRIBUTE WHICH SETS OR RETURNS THE DESTINATION STATION (DESTUNITF).. THIS ATTRIBUTE MAY ONLY BE SET BY AN MCS; AND WHEN SET, THE DESTINATION MCS NUMBER (DSTMCSF) WILL BE SET USING THE NUMBER ASSOCIATED WITH THE MCS SETTING THIS ATTRIBUTE.

45 SOURCESTATION INTEGER 2

THIS IS AN INTEGER VALUED ATTRIBUTE WHICH SETS OR RETURNS THE ORIGINATION STATION (ORGUNITF). THIS ATTRIBUTE MAY ONLY BE SET BY AN MCS. (SEE DESTSTATION.)

47 SOURCEKIND INTEGER 1

THIS IS AN INTEGER VALUED ATTRIBUTE WHICH RETURNS THE UNIT TYPE ASSOCIATED WITH THE ORIGINATING UNIT. FOR EXAMPLE:

CARDREADER = 9, REMOTE = 3, SPO = 2

49 MAXWAIT REAL 2

THIS IS A REAL VALUED ATTRIBUTE WHICH SPECIFIES THE MAXIMUM NUMBER OF SECONDS A TASK CAN AFFORD TO WAIT ON SPECIFIED SYSTEM FUNCTIONS, SUCH AS DMSII RECORD LOCKING.

CLARIFIED TASK ATTRIBUTES

22 OPTION REAL 2

IF THE BDBASE BIT IN OPTION (BIT SIX) IS SET ON AN ACTIVE TASK (FOR EXAMPLE, "MYSELF.OPTION:=64") THEN THE BACKUP FILES WILL NOT BE AUTOMATICALLY PRINTED. THE PROBLEM IS THAT IT IS TOO LATE TO SET UP A JOBFILE FOR THAT TASK.

NO ERROR IS GIVEN BECAUSE THE ABILITY TO SET BDBASE ON AN ACTIVE TASK IS USEFUL WHEN USED IN CONJUNCTION WITH BACKUPPREFIX TO CREATE BD FILES WHICH WILL BE PROGRAMMATICALLY PRINTED.

D0581 MCP - TASK ATTRIBUTES - 11-18-73

29 BACKUPPREFIX POINTER 0

THIS ATTRIBUTE ALLOWS A PREFIX OTHER THAN THE DEFAULT "BD" TO BE USED AS THE MULTI-FILE ID OF THE TASK"S BACKUP FILES. ANY VALID FILE TITLE MAY BE SPECIFIED AS THIS PREFIX, AND THIS ATTRIBUTE HAS EFFECT ONLY IF BIT SIX OF THE TASK"S OPTION WORD (BDBASE) IS ON. THE CONSTRUCT IS USEFUL WHEN A PROCESS OTHER THAN AUTOPRINT IS TO PRINT THE BACKUP FILES THUS PREVENTING THE AUTOMATIC PRINTING AND REMOVAL OF THE FILES.

IF A BACKUP PREFIX IS USED BY A TASK RUNNING UNDER A USERCODE, THE RESULTING FILE WILL BE PLACED UNDER ITS USER DIRECTORY.

TO SET BACKUPPREFIX TO NAMES OF THE FORM (C)B OR *B (WHERE C IS A DIFFERENT USERCODE THAN THAT BEING RUN UNDER), THE PROGRAM MUST BE PRIVILEGED.

D0593 MCP - SETSTATUS LOGGING - 12-06-73

THIS PATCH IMPLEMENTS SETSTATUS LOGGING (I.E., CONSOLE COMMANDS WHICH GENERATE A CALL ON THE MCP PROCEDURE SETSTATUS). LOGGED IN SYSTEM/SUMLOG ARE THE ARRAYS WHICH ARE PASSED TO SETSTATUS FOR ANALYSIS.

THOSE COMMANDS WHICH ARE MIX ORIENTED (XS, ST, DS, OF, OK, FR, RM, DP, PR, HI, OG, AX, IL, UL, OU, FM, US, LJ) ARE ALSO LOGGED INTO THEIR APPROPRIATE JOB LOGS AND PRINTED ON THE THE JOB SUMMARY.

D0615 MCP - CM ON THREE CPU SYSTEMS - 01-27-74

CM WILL SOMETIMES FAIL ON A THREE PROCESSOR SYSTEM. THE SYMPTOMS OF THIS FAILURE ARE EITHER A SYSTEM HANG JUST AFTER THE CM OR THE LOSS OF ONE OR MORE PROCESSORS WHEN THE SYSTEM HAS INITIALIZED. A HALT/LOAD WILL RECOVER THE SYSTEM PROPERLY.

D0616 MCP - MEMDUMP IMPROVEMENTS - 12-28-73

THE MCP NOW PRINTS THE UNIT NUMBER AND SERIAL NUMBER OF THE TAPE WHERE IT TAKES A DUMP. FURTHER, IF THE TAPE REQUIRES A SERIAL NUMBER (BECAUSE IT IS UNLABELED OR IN ERROR STATE), THE OPERATOR IS SO INFORMED. HE THEN HAS TWO OPTIONS: HE MAY RETRANSMIT THE MT OR TRANSMIT A PG COMMAND, AND A SERIAL NUMBER OF "000000" WILL BE USED. HE MAY ALSO USE THE REGULAR FORM OF THE "SN" COMMAND (SEE D0635 FOR SN CHANGES).

WHEN A MEMORY DUMP IS TAKEN THE NAME OF THE PROCESS WILL BE DISPLAYED FOLLOWING THE DUMP BY INFO.

EXAMPLE

```
DUMP BY MEMORY TESTOR @0005:142F:5
PROCESSID:  DISKMAPPER
```

D0618 MCP - OVERHEADCHARGED OPTION - 12-28-73

THIS COMPILE-TIME OPTION, WHEN RESET FOR MULTIPROCESSOR SYSTEMS, WILL TEND TO REDUCE THE MIX DEPENDENCY OF PROCESS TIMES BY INHIBITING PROCESS TIME CHARGING WHEN A TASK IS INTERRUPTED TO PERFORM A STACKSEARCH. WHEN THE OPTION IS SET THE SYSTEM WILL CHARGE AS BEFORE.

D0619 MCP - SETSTATUS AND GETSTATUS - SB - 12-28-73

SETSTATUS

THE "SB" CALL ON SETSTATUS IS IMPLEMENTED ON 2.6 BY A CALL WITH TYPE=MISCELLANEOUS REQUEST (2) AND SUBTYPE=25. VALUE=0.

THE SB CALL SPECIFIES WHAT FORM OF "SUBSTITUTE BACKUP" EQUATION IS TO BE USED.

THE ARRAY PASSED MUST BE AT LEAST EIGHT WORDS LONG. ONLY EIGHT WORDS WILL BE USED. WORD ZERO MUST HAVE THE VALUE EIGHT AS THE CALL HAS SIX WORDS OF ACTUAL INFORMATION PLUS THE TWO PARAMETER WORDS AT THE START. THE SECOND PARAMETER WORD (WORD ONE) MUST CONTAIN 0&6[38:6]. WORDS TWO THROUGH SEVEN CONTAIN SB INFORMATION IN THE FORM OF ANY COMBINATION OF THE EBCDIC BYTES (4"41" FOR BACKUP DISK, 4"4D" FOR BACKUP TAPE7, 4"4E" FOR BACKUP TAPE9, 4"4F" FOR BACKUP PETAPE, 4"51" FOR BACKUP PACK, 4"5C" FOR BACKUP TAPE). THE ORDER OF PREFERENCE IS FROM RIGHT TO LEFT WITHIN EACH WORD. 4"00" IS ALSO LEGAL AS A DELIMITER FOR THE END OF STRING.

EACH WORD FROM TWO TO SEVEN REPRESENTS A DIFFERENT BACKUP MEDIUM AND THE SUBSTITUTIONS ARE CONTAINED IN EACH WORD.

WORD 2 IS BACKUP PACK
WORD 3 IS BACKUP TAPE7
WORD 4 IS BACKUP TAPE9
WORD 5 IS BACKUP PETAPE
WORD 6 IS BACKUP TAPE
WORD 7 IS BACKUP DISK

IF ONE OF THE SIX LEGAL EBCDIC BYTES (AS OPPOSED TO 4"00") OCCURS IN A SINGLE WORD, OVERHEAD IS INTRODUCED. SETSTATUS FLAGS AS AN ERROR ANY OCCURRENCE OF A EBCDIC BYTE OTHER THAN THE SIX LEGAL ONES PLUS THE DELIMITER 4"00" WITHIN THE WORDS TWO THROUGH SEVEN.

GETSTATUS

GETSTATUS RETURNS THE SIX WORDS OF SBINFO AS SETSTATUS ENTERED THEM. SEE THE SETSTATUS DISCUSSION FOR THE PARTICULARS.

THE SPECIFIC GETSTATUS CALL IS GETSTATUS(0&MISCREQUEST GSTYPEF & 1 GSSUBTYPEF, 0, 0 & SET(GSSBINFO), ARRAYPASSED) WHERE MISCREQUEST IS TWO, GSTYPEF IS [7:8], ASSUBTYPEF IS [15:8], AND 0 & SET(GSSBINFO) IS EQUAL TO 4"1000000".

D0621 MCP - MEMORY INTEGRAL CALCULATION - 12-28-73

ON 2.6, MEMORY INTEGRALS FOR TASKS ARE DETERMINED IN THE FOLLOWING MANNER:

FOR DATA, EACH TIME AN ADDITIONAL DATA AREA IS MADE PRESENT (A FORGOTTEN OR OVERLAYED ONE), THE DATA MEMORY INTEGRAL IS INCREASED (DECREASED) BY THE SIZE OF THE AREA TIMES THE SUM OF THE I/O AND PROCESSOR TIME SINCE THE LAST ADDITION. THIS RESULTS IN A DATA MEMORY INTEGRAL WHICH IS THE ACCUMULATION, FOR ALL AREAS, OF THE SIZE OF EACH AREA TIMES THE SUM OF I/O AND PROCESSOR TIME DURING WHICH THIS MEMORY AREA WAS PRESENT.

FOR CODE, THE MEMORY INTEGRAL OF A TASK IS THE AVERAGE SIZE OF ITS CODE AREA TIMES THE SUM OF ITS PROCESSOR AND I/O TIMES. THE AVERAGE SIZE OF A TASK"S CODE AREA IS THE MEMORY ALLOCATED TO ITS D1 STACK AVERAGED OVER THE ELAPSED TIME ITS D1 STACK HAS BEEN ACTIVE. IN THE CASE OF REENTRANT CODE, SEVERAL TASKS MAY SHARE THE SAME D1 STACK, MAKING AN ACCURATE COMPUTATION OF EACH TASK"S CODE MEMORY INTEGRAL IMPOSSIBLE. HOWEVER, A REASONABLY ACCURATE CODE MEMORY INTEGRAL MAY BE OBTAINED BY USING THE PREVIOUSLY STATED ALGORITHM.

FOR INDEPENDENT RUNNERS, SUCH AS LIBRARY MAINTENANCE, THE D1 STACK IS THE MCP STACK. SINCE IT IS UNREASONABLE TO CHARGE A USER FOR ALL THE CODE IN USE BY THE MCP, A ZERO IS ENTERED IN THE CODE MEMORY INTEGRAL AND AVERAGE CODE MEMORY USAGE FOR AN INDEPENDENT RUNNER.

AVERAGE CORE UTILIZATION MAY BE OBTAINED BY DIVIDING THE CORE INTEGRAL BY THE SUM OF THE I/O AND PROCESSOR TIMES. THE AVERAGE IS NOW LOGGED AND PRINTED ON THE JOB SUMMARY.

NOTE THAT THESE INTEGRALS ARE CORE IN USE. OVERLAID AREAS ARE NOT INCLUDED, MAKING THE RESULTS MIX DEPENDENT.

THESE ALGORITHMS ARE SUBJECT TO CHANGE IN FUTURE RELEASES.

D0628 MCP - JOBS WITH PUNCH OUTPUT - 01-09-73

IF A JOB PRODUCES PUNCH OUTPUT, ITS JOBFILE MAY NOT BE REMOVED UNTIL ITS PUNCH OUTPUT HAS BEEN PUNCHED. THE JOBFILE MUST BE KEPT IN CASE THE SITE HAS A PERIPHERAL ASSOCIATION SET UP FOR THE ORIGINATING UNIT. IF THE SYSTEM HAS NO PUNCH, THE ONLY WAY TO ELIMINATE SUCH A JOBFILE IS TO DO AN EP.

D0629 MCP - EXTERNAL DECKS IN WFL - 01-09-73

UNLIKE DECKS INTERNAL TO A WFL JOB, ALL EXTERNAL DECKS MUST TERMINATE WITH <I> END. OTHERWISE THE FIRST CARD OF THE FOLLOWING DECK WILL BE GIVEN TO THE FILE OPENING THE FIRST DECK AND WILL NOT BE FOUND WHEN THE SECOND DECK IS SOUGHT.

D0670 MCP - MYJOB - 01-20-74

THIS PATCH IMPLEMENTS MYJOB, A TASK VALUED MCP PROCEDURE WHICH RETURNS THE JOB OF THE RUNNING STACK, OR, IF THE RUNNING STACK HAS NO JOBSTACK, THEN "MYSELF".

D0717 MCP - WFL PRINTOUT ON SIMPLE STMTS - 02-20-74

NO JOBFILE IS CREATED FOR SIMPLE DISK DIRECTORY MAINTENANCE STATEMENTS SUCH AS REMOVE. THE REASON FOR NOT CREATING A JOBFILE IS TO PERMIT THE OPERATOR TO RECOVER FROM A NO-USER-DISK SITUATION. IT IS PARTICULARLY IMPORTANT TO BE ABLE TO DO THIS FROM THE CARD READER SINCE UNDER CERTAIN RARE CONDITIONS THE CONTROLLER MAY BE HUNG UP.

TO INSURE PRINTOUT, USERS NEED ONLY MAKE THEIR STATEMENTS SLIGHTLY MORE COMPLEX. FOR EXAMPLE:

D0717 MCP - WFL PRINTOUT ON SIMPLE STMTS - 02-20-74

<I> JOB X; BEGIN L: CHANGE A/B TO C/D; END JOB

D0724 MCP - I-O ERROR DETECTION - 11-18-73

THE DISK SUBSYSTEM IS CAPABLE OF GIVING A RESULT WITH NO ERRORS EVEN THOUGH IT DID NOT TRANSFER THE ENTIRE LENGTH REQUESTED. A TEST HAS BEEN INCLUDED UNDER THE DISKCHECK COMPILE TIME OPTION TO DETECT THIS CONDITION AND REPORT AN ERROR OF "D" FOR IT (BUSY AND NOT-READY).

THIS OPTION HAS BEEN RELEASED SET ON THE 2.6 RELEASE. SINCE THIS INVOKES AN OVERHEAD COST, USERS HAVING NO DISK PROBLEMS MAY WISH TO RESET THIS OPTION.

NETWORK DEFINITION LANGUAGE (NDL)

D0560 NDL - NDL SCANNER PROCEDURES - 10-28-73

THE FOLLOWING CHANGES HAVE BEEN MADE IN THE NDL COMPILER:

1. STRING DEFINITION

   NDL ALLOWS USERS TO CONCATENATE <CONSTANT IDENTIFIER>"S WITH STRINGS. THE NEW SYNTAX IS DEFINED AS FOLLOWS:

   <STRING>::= <SIMPLE STRING> / <UNITARY STRING> / <CONSTANT IDENTIFIER> / <STRING><STRING>

   EXAMPLE:

   ```
   TRANSMIT SOH STX 4"00".
   ```

   <SIMPLE STRING>"S, <UNITARY STRING>"S AND <CONSTANT IDENTIFIER>"S, MUST STILL BE MULTIPLES OF EIGHT BITS TO BE CONSIDERED VALID.

2. DOLLAR OPTION RECOGNITION

   DOLLAR OPTIONS THAT HAVE THE SAME FIRST FIVE CHARACTERS AND IDENTICAL CHARACTER COUNTS ARE EQUIVALENT.

   EXAMPLE:

   TWO EQUIVALENT STATEMENTS:

   ```
   $ SET SYNTAX
   $ SET SYNTAA
   ```

3. DOLLAR CARDS IN THE SYMBOLIC

   CARDS WITH A DOLLAR SIGN IN COLUMN TWO WILL BE OUTPUT TO THE FILE "NEWTAPE".

4. ADDITIONAL DOLLAR OPTIONS

THE FOLLOWING DOLLAR OPTIONS HAVE BEEN ADDED:

A. LISTP - RECORDS FROM THE FILE "CARD" ARE LISTED ON THE FILE "LINE".

B. SEQERR - IF THE FILE "TAPE" CONTAINS SEQUENCE ERRORS THAT HAVE NOT BEEN VOIDED, THEY ARE FLAGGED AND THE FILES "NIF" AND "DCPCODE" ARE NOT LOCKED.

C. NEWSEQERR - IF "NEW" IS SET AND THE FILE "NEWTAPE" CONTAINS SEQUENCE ERRORS, THEY ARE FLAGGED AND THE FILES "NIF", "DCPCODE" AND "NEWTAPE" ARE NOT LOCKED.

D. ERRLIST - ALL ERRORS ARE FLAGGED ON THE FILE "ERRORFILE", AS WELL AS THE FILE "LINE".

E. PAGE - THE FILE "LINE" SKIPS TO THE TOP OF FORM.

F. LIMIT - THE MAXIMUM NUMBER OF ERRORS IS SET TO THE NUMBER FOLLOWING LIMIT. DEFAULT IS 10 FROM CANDE AND 150 OTHERWISE.

G. VERSION - VERSION IS SET TO THE FOLLOWING SEQUENCE OF DIGITS AND PERIODS. THIS VERSION NUMBER (UP TO SIX CHARACTERS) IS COMBINED WITH A PATCH NUMBER APPEARING IN COLUMN 80 OF THE INPUT RECORD AND IS PUT IN COLUMN 80 OF THE FILE "NEWTAPE" RECORD.

H. $ - PRINT ALL DOLLAR OPTION CARDS ON THE FILE "LINE".

D0561 NDL - LIMIT MAXIMUM MCS"S - 10-28-73

CHANGES TO THE MCP HAVE LIMITED THE MAXIMUM NUMBER OF MCS"S. NDL WILL GIVE A SYNTAX ERROR IF MORE THAN 47 ARE DEFINED.

D0575 NDL - CRC[0],CRC[1] AS BYTE VARIABLE - 11-18-73

NDL ALLOWS USERS TO ACCESS THE TWO BYTE CRC. THE MOST SIGNIFICANT BYTE IS DESIGNATED CRC[0] AND THE LEAST SIGNIFICANT BYTE IS CRC[1]. IF THESE BYTES ARE USED WITH INCOMPATIBLE PARITY, A WARNING WILL BE ISSUED. BCC AND CRC[0] MAY NOT BE USED IN THE SAME REQUEST SET.

D0622 NDL - CHANGE MSG SPACE ALLOCATION - 12-28-73

THE USER IS NOW ABLE TO ALLOCATE AN INITIAL AMOUNT OF MESSAGE SPACE EXPLICITLY. THIS DCP SECTION STATEMENT HAS THE FOLLOWING SYNTAX:

<TERMINAL LIST>::= <TERMINAL IDENTIFIER> <OPTIONAL MSGSPACE> /
<TERMINAL LIST>,<TERMINAL LIST>
<OPTIONAL MSGSPACE>::= (MSGSPACE=<INTEGER>)

FOR EXAMPLE

TERMINAL = T2741(MSGSPACE=3),TELETYPE(MSGSPACE=4).

FOR TERMINALS OF THE SAME MESSAGE SPACE SIZE, THE ESTIMATES ARE ADDED. IF NO ESTIMATE FOR A SIZE IS SPECIFIED, TWO SPACES ARE ALLOCATED. THE DOLLAR OPTION SUMMARY CAN STILL BE USED TO SEE WHAT SPACE HAS BEEN ALLOCATED.

D0667 NDL - ADD DOLLAR OPTION NOWWARN - 01-20-74

THE DOLLAR OPTION "NOWARN" HAS BEEN ADDED. WHEN SET, ALL COMPILER WARNINGS ARE SUPPRESSED. THE DEFAULT VALUE IS FALSE.

D0726 NDL - READ-ONLY DCPCODE FILES ON 2.7 - 03-03-74

THE NDL COMPILER ON THE 2.7 SYSTEM RELEASE WILL SET THE FILEKIND OF THE DCPCODE FILE TO A NEW VALUE (TO BE SPECIFIED) WHICH WILL

DISALLOW PROGRAMS FROM WRITING INTO THE FILE. THIS IS BEING DONE TO PRESERVE THE INTEGRITY OF DCPCODE FILES.

PACKDIRECTORY (PACKDIR)

D0668 PACKDIR - NO-FILE TERMINATES NORMALLY - 01-20-74

THE REQUEST OF A PACKDIR FOR A NON-EXISTENT PACK OR FILE ON A PACK WILL NO LONGER RETURN A PACK-REQUIRED MESSAGE. INSTEAD AN ERROR MESSAGE WILL BE PRINTED AND THE PROGRAM WILL TERMINATE NORMALLY.

D0669 PACKDIR - REVISE PARAMETER SCANNED - 01-20-74

THIS PATCH ALLOWS THE "NAME =" PARAMETER TO HAVE IMBEDDED BLANKS. PREVIOUSLY, A FILE ATTRIBUTE ERROR #39 WOULD RESULT.

D0671 PACKDIR - FLAG MISLEADING RESULTS - 01-06-74

MISLEADING RESULTS IN THE REPORT DEALING WITH AREAS WHICH CAN BE MADE AVAILABLE BY REMOVING ONE FILE ARE NOW FLAGGED. THESE PROBLEMS ARE THE RESULT OF FILE CONFLICTS.

D0672 PACKDIR - IDENTIFY SYSTEM IN HEADING - 01-06-74

THE SYSTEM NUMBER OF THE COMPUTER ON WHICH PACKDIR IS RUNNING IS NOW INCLUDED IN THE PAGE HEADINGS.

PATCH

D0613 PATCH - $GOTO WITH VOIDT - 01-27-74

THIS PATCH CAUSES AN ERROR MESSAGE TO BE PRINTED WHEN A $GOTO CARD IS ENCOUNTERED WHILE VOIDT IS SET. THE ACTION TAKEN BY THE COMPILERS IN THIS CASE IS TO IGNORE THE GOTO. THIS SITUATION PREVIOUSLY CAUSED SYSTEM/PATCH TO GO INTO A LOOP.

D0634 PATCH - PATCH DECKS ON DISK FILES - 01-09-73

THIS PATCH IMPLEMENTS THE OPTION PATCHDECK, WHICH IS RECOGNIZED BY SYSTEM/PATCH TO MEAN THAT THE CURRENT PATCH DECK IS ON A DISK FILE INSTEAD OF ON CARDS. PATCHDECK REPLACES, AND IS IDENTICAL TO THE DISK $ SPECIFICATION. THE RATIONALE FOR THE CHANGE IS THAT THE FUNCTIONING OF THE DISK AND DISK $ SPECIFICATIONS IS DIFFERENT IN SEVERAL RESPECTS, BUT THE SIMILARITY IN SYNTAX DOES NOT MAKE THIS APPARENT AND COULD LEAD TO CONFUSION.

THE BASIC DIFFERENCE BETWEEN SPECIFYING DISK AND PATCHDECK IS THAT THE LATTER SPECIFICATION WILL CAUSE MARK NUMBERS TO BE PUT ON THE PATCH CARDS (PROVIDED MARK IS SET) WHEREAS THE FORMER ASSUMES THAT THE CARD IMAGES ALREADY CONTAIN MARK NUMBERS. HOWEVER, THERE IS A BASIC DIFFERENCE IN THE WAY THE CARD IMAGES ARE PROCESSED. DISK ASSUMES THAT THE FILE WAS CREATED BY SYSTEM/PATCH AND THEREFORE CONTAINS NO SEQUENCE ERRORS AND NO $ CARDS WHICH IT MUST PROCESS ($ VOIDT, SEQ, ETC.). NO CHECKING IS DONE, AND IF THE ABOVE CONDITION IS NOT MET, UNPREDICTABLE RESULTS MAY OCCUR. PATCHDECK, ON THE OTHER HAND, TREATS THE DISK FILE EXACTLY LIKE A CARD DECK AND THEREFORE DOES NOT HAVE THE ABOVE RESTRICTIONS. THE FOLLOWING IS AN EXAMPLE OF A SYSTEM/PATCH DECK USING A PATCHDECK CARD (NOTE: DISK $ SPECIFICATION IS STILL ALLOWED IN ORDER TO MAINTAIN COMPATIBILITY WITH PREVIOUS VERSIONS OF PATCH).

```
$.SET MARK COMPARE
$#PATCH 001
$. PATCHDECK PATCH/FILE
$#PATCH 002
$.PATCHDECK (ABCD)PATCHES
```

THIS PATCH ALSO CORRECTS MINOR PROBLEMS IN THE HANDLING OF FILENAMES WHICH ARE PRECEDED BY USERCODES ON PATCHDECK AND DISK CARDS.

PLI

D0649 PLI - SORT STATEMENT - 10-28-73

FUNCTION:

THE SORT STATEMENT CAUSES A SET OF RECORDS TO BE SORTED ACCORDING TO THE OPTIONS SPECIFIED IN THE SORT STATEMENT.

GENERAL FORMAT:

```
<SORT-STATEMENT>::= SORT <SORT IDENTIFIER> [ON] <SORT OPTIONS>
<SORT OPTIONS>::= <KEY OPTION> [<INPUT OPTION>] [<OUTPUT OPTION>]
                  [<MEMORY OPTION>]
<KEY OPTION>::= ASCENDING [KEY] (<IDENTIFIER>...) /
                DESCENDING [KEY] (<IDENTIFIER>...) / ...
<INPUT OPTION>::= USING FILE (<FILE EXPRESSION>) /
                  INPUT (<ENTRY CONSTANT>)
<OUTPUT OPTION>::= GIVING FILE (<FILE EXPRESSION>) /
                   OUTPUT (<ENTRY CONSTANT>)
<MEMORY OPTION>::= ENVIRONMENT (TAPES=<CONSTANT EXPRESSION>,
                   CORESIZE=<CONSTANT EXPRESSION>,
                   DISKSIZE=<CONSTANT EXPRESSION>)
```

SYNTAX RULES:

1. THE <SORT OPTIONS> MAY APPEAR IN ANY ORDER.
2. THE <SORT IDENTIFIER> MUST BE AN AGGREGATE WHICH DESCRIBES THE INDIVIDUAL RECORDS TO BE SORTED AND MAY NOT BE CONTROLLED OR BASED.

GENERAL RULES:

1. THE <KEY OPTION> MUST APPEAR, AS THIS OPTION SPECIFIES THE ORDER

IN WHICH THE RECORDS ARE TO BE SORTED AND THE KEYS TO BE USED IN THE SORT. THE ORDER OF PRECEDENCE OF THE KEYS IS DETERMINED BY THE ORDER OF APPEARANCE OF THE KEY IN THE <KEY OPTION>. THE SORT KEYS ARE SUBJECT TO THE FOLLOWING RULES:

A. EACH KEY MUST BE DEFINED ON THE <SORT IDENTIFIER>.

B. NO VARIABLE LENGTH KEYS ARE ALLOWED.

C. THE RECORDS MUST ALL BE OF SOME FIXED LENGTH, AND THE KEYS MUST BE IN THE SAME LOCATION IN EACH RECORD.

2. THE <INPUT OPTION> MAY BE EITHER A FILE DESIGNATION OR AN INPUT PROCEDURE DESIGNATION. IF THE <INPUT OPTION> IS NOT EXPLICITLY STATED, THE <INPUT OPTION>

   USING FILE (SYSIN)

   IS ASSUMED. IF AN EXPLICIT FILE DESIGNATION IS USED, THE FILE SHOULD BE DECLARED AS AN INPUT FILE. THE RECORDS OF THE FILE ARE THEN USED AS INPUT TO THE SORT. THE SORT REQUIRES THAT AN INPUT FILE PASSED TO THE SORT BE CLOSED PRIOR TO THE CALL TO THE SORT. IF AN INPUT PROCEDURE IS USED, HOWEVER, THE PROCEDURE MUST HAVE SORTINPUT DECLARED IN THE <OPTIONS-LIST> OF THE PROCEDURE DECLARATION. THE INPUT PROCEDURE IS SUBJECT TO THE FOLLOWING RULES:

   A. THE INPUT PROCEDURE MUST HAVE ONE AND ONLY ONE PARAMETER. THIS PARAMETER MUST BE DECLARED CHAR (*).

   B. THE INPUT PROCEDURE MUST RETURN A BIT (1) VALUE.

   C. A FALSE VALUE ("0"B) MUST BE RETURNED BY THE INPUT PROCEDURE UNTIL THE END OF THE INPUT DATA IS ENCOUNTERED. THEN, A TRUE VALUE ("1"B) MUST BE RETURNED. (NOTE: THE DOUBLE QUOTE HAS BEEN USED FOR THE SINGLE QUOTE IN THE ABOVE.)

   D. AS LONG AS A FALSE VALUE IS BEING RETURNED, THE INPUT PROCEDURE SHOULD INSERT THE NEXT RECORD TO BE SORTED INTO ITS PARAMETER.

3. THE <OUTPUT OPTION> MAY BE EITHER A FILE DESIGNATION OR AN OUTPUT PROCEDURE DESIGNATION. IF THE <OUTPUT OPTION> IS NOT EXPLICITLY STATED, THE <OUTPUT OPTION>

   GIVING FILE (SYSPRINT)

   IS ASSUMED. IF A FILE DESIGNATION IS USED, THE FILE SHOULD BE DECLARED AS AN OUTPUT FILE. THE SORT WILL THEN WRITE THE SORTED OUTPUT TO THIS FILE. AS WITH AN INPUT FILE, AN OUTPUT FILE MUST BE CLOSED PRIOR TO THE CALL TO THE SORT. IF AN OUTPUT PROCEDURE IS USED, HOWEVER, THE PROCEDURE MUST HAVE SORTOUTPUT DECLARED IN THE <OPTIONS-LIST> OF THE PROCEDURE DECLARATION. THE OUTPUT PROCEDURE IS SUBJECT TO THE FOLLOWING RULES:

   A. THE OUTPUT PROCEDURE MUST HAVE TWO PARAMETERS. THE FIRST PARAMETER MUST BE DECLARED AS BIT (1), AND THE SECOND PARAMETER MUST BE DECLARED CHAR (*).

   B. THE FIRST PARAMETER WILL CONTAIN A FALSE VALUE ("0"B) AS LONG AS THE SECOND PARAMETER CONTAINS A SORTED RECORD. WHEN ALL RECORDS HAVE BEEN RETURNED TO THE OUTPUT PROCEDURE BY SORT, THE FIRST PARAMETER WILL CONTAIN A TRUE VALUE ("1"B), AND THE SECOND PARAMETER MUST NOT BE ACCESSED. (NOTE: THE DOUBLE QUOTE HAS BEEN USED FOR THE SINGLE QUOTE IN THE ABOVE.)

4. THE <MEMORY OPTION> SPECIFIES THE NUMBER OF TAPES TO BE USED BY THE SORT AS WELL AS THE CORE SIZE AND THE DISK SIZE TO BE ALLOCATED FOR THE SORT. THE OPTIONS MAY APPEAR IN ANY ORDER, AND ANY OF THE OPTIONS MAY BE DELETED. THE FOLLOWING DEFAULT VALUES ARE ASSUMED FOR ANY OPTION NOT EXPLICITLY STATED:

   TAPES=3, CORESIZE=12000, DISKSIZE=600000

   THE NUMBER OF TAPES SHOULD BE LESS THAN OR EQUAL TO EIGHT AND GREATER THAN OR EQUAL TO THREE.

EXAMPLES:

CONSIDER THE FOLLOWING DECLARATIONS: (NOTE: THE DOUBLE QUOTE HAS

BEEN USED FOR THE SINGLE QUOTE IN THE FOLLOWING EXAMPLE.)

```
DCL F FILE INPUT RECORD ENV (KIND="READER", MAXRECSIZE=80),
    G FILE OUTPUT RECORD ENV (KIND="PRINTER", MAXRECSIZE=132);
DCL AR(10);
DCL CH CHAR(10),
    KEYS CHAR(5) DEFINED CH POSITION (6);
SORTIN:  PROC(A)  RETURNS(BIT(1))
    OPTIONS (SORTINPUT);
    DCL A CHAR(*);
        .
        .
        .
    END SORTIN;
DCL  1 AS,
       2 B CHAR(1),
       2 C CHAR(10);
SORTOUT:  PROC (B,A)  OPTIONS (SORTOUTPUT);
       DCL  B  BIT(1),
            A  CHAR(*);
          .
          .
          .
       END SORTOUT;
```

THE FOLLOWING <SORT-STATEMENT>"S WOULD BE LEGAL USING THE ABOVE DECLARATIONS:

1.
```
SORT  AS ON
      DESCENDING KEY (AS.B AS.C)
      USING FILE (F) OUTPUT (SORTOUT)
      ENV (TAPES=4, CORESIZE=30000, DISKSIZE=100000);
```

2.
```
SORT  AR
      INPUT (SORTIN)
      ASCENDING KEY (AR(1)) DESCENDING (AR(6))
      ENV (CORESIZE=25000, TAPES=5);
```

3.
```
SORT  CH
```

D0649 PLI - SORT STATEMENT - 10-28-73

```
GIVING FILE (G)
ASCENDING (KEYS);
```

4. SORT CH ON ASCENDING (KEYS);

D0650 PLI - MYSELF TASK - 11-18-73

THE TASK MYSELF (THE TASK ASSOCIATED WITH A JOB) IS IMPLEMENTED AS A TASK DESIGNATOR FOR SYSTEM TASK ATTRIBUTE ASSIGNMENT AND REFERENCE AS A BUILT-IN FUNCTION. FOR MYSELF TO BE RECOGNIZED AS A TASK IT MUST BE DECLARED TEMPORARILY AS BUILTIN OR BE FOLLOWED BY EMPTY PARENTHESES IN AT LEAST ONE REFERENCE IN A PROGRAM.

FOR THE SYNTAX OR TASK ATTRIBUTE ASSIGNMENT AND REFERENCE STATEMENTS, REFER TO PL/I D0653.

D0651 PLI - FILE DECLARATIONS - 11-18-73

THE FOLLOWING IS A LIST OF SYSTEM FILE ATTRIBUTE SPECIFICATIONS GENERATED BY THE COMPILER. THEY MAY BE OVERRIDDEN BY SPECIFICATIONS INSIDE THE ENVIRONMENT FOR A FILE.

1. SAVEFACTOR IS SET TO 30
2. UNITS IS SET TO CHARACTERS
3. IF A FILE IS EXTERNAL AND HAS THE NAME SYSPRINT:
   A. KIND IS SET TO REMOTE IF COMPILED THROUGH CANDE, OTHERWISE TO PRINTER
   B. MYUSE IS SET TO OUT
   C. PAGESIZE IS SET TO 58
   D. MAXRECSIZE IS SET TO 72 IF COMPILED THROUGH CANDE, OTHERWISE TO 132.
4. IF A FILE IS EXTERNAL AND HAS THE NAME SYSIN:

A. KIND IS SET TO REMOTE IF COMPILED THROUGH CANDE, OTHERWISE TO READER

B. MYUSE IS SET TO IN

C. MAXRECSIZE IS SET TO 72 IF COMPILED THROUGH CANDE, OTHERWISE TO 80.

5. IF A FILE IS DECLARED PRINT:

A. KIND IS SET TO PRINTER

B. MYUSE IS SET TO OUT

C. PAGESIZE IS SET TO 58

D. MAXRECSIZE IS SET TO 132

6. IF A FILE IS NOT DECLARED PRINT, BUT INSIDE THE ENVIRONMENT KIND IS SET TO PRINTER:

A. MAXRECSIZE IS SET TO 132

B. MYUSE IS SET TO OUT

C. PAGESIZE IS SET TO 58

7. IF INSIDE THE ENVIRONMENT, KIND IS SET TO READER

A. MAXRECSIZE IS SET TO 80

B. MYUSE IS SET TO IN

8. IF A FILE IS DECLARED INPUT, MYUSE IS SET TO IN

9. IF A FILE IS DECLARED OUTPUT, MYUSE IS SET TO OUT

10. IF A FILE IS DECLARED UPDATE, MYUSE IS SET TO I/O, UPDATEFILE IS SET TO TRUE

THE ABOVE IS ALSO THE ORDER IN WHICH THE ATTRIBUTES ARE SET. IN CASE OF CONFLICTS THE LAST SPECIFICATION WILL APPLY. FOR EXAMPLE:

```
DCL SYSPRINT INPUT
```

THE MYUSE ATTRIBUTE IS SET TO IN.

D0651 PLI - FILE DECLARATIONS - 11-18-73

THE FOLLOWING ATTRIBUTE SPECIFICATIONS GENERATED BY THE COMPILER MAY NOT BE OVERRIDDEN BY SPECIFICATIONS WITHIN THE ENVIRONMENT:

1. INTMODE IS SET TO EBCDIC
2. IF A FILE IS DECLARED DIRECT OR KEYED, THE DIRECT SYSTEM FILE ATTRIBUTE IS SET TO TRUE.

D0652 PLI - STMTNO OPTION - 11-18-73

THIS PATCH CHANGES THE STMTNO OPTION INFORMATION. IF THERE ARE NO SEQUENCE NUMBERS ON THE INPUT CARD OR TAPE FILES THE CARD NUMBER WILL BE CARRIED INSTEAD OF A SEQUENCE NUMBER, FOR EXAMPLE:

```
002:0003:4 (00006:00002400) IF SEQUENCE NUMBER
002:0003:4 (00006:CARD 00011) IF NO SEQUENCE NUMBER
```

SYSTEM TASK ATTRIBUTE WILL NO LONGER BE RECOGNIZED.

SYSTEM TASK ATTRIBUTES

```
<INTEGER VALUED SYSTEM TASK ATTRIBUTE>::= STACKNO / CORRESTIMATE /
      DECLAREDPRIORITY / MAXPROCTIME / MAXIOTIME / TARGETTIME /
      STACKSIZE / OPTION / RESTART / SUBSPACES / DECKGROUPNO /
      CLASS / COMPILETIME / ORGUNIT / MAXCARDS / MAXLINES /
      JOBNUMBER

<REAL VALUED SYSTEM TASK ATTRIBUTE>::= TASKVALUE / HISTORY /
      TYPE / STOPPOINT / INITIATOR / STATION / TASKATTERR /
      SOURCESTATION / DESTSTATION / SOURCEKIND
```

<BOOLEAN VALUED SYSTEM TASK ATTRIBUTE>::= LOCKED / VALIDITYBITS

<STRING VALUED SYSTEM TASK ATTRIBUTE>:= NAME / USERCODE / FILECARDS / BACKUPPREFIX / STACKHISTORY / CHARGECODE / DESTNAME

MNEMONICS FOR SYSTEM TASK ATTRIBUTES:

THE FOLLOWING IS A LIST OF SYSTEM TASK ATTRIBUTES THAT HAVE MNEMONICS AND THE MNEMONICS THAT APPLY TO THEM.

ATTRIBUTE	MNEMONICS
TYPE	PROCESS, CALL, RUN, JOBSTACK
LOCKED	TRUE, FALSE
VALIDITYBITS	TRUE, FALSE
COMPILETIME	COMPILEANDGO, SYNTAX, LIBRARYANDGO, LIBRARY NEVERUSED, SCHEDULED, ACTIVE, SUSPENDED, TERMINATED, BADINITIATE
OPTION	LONG, FAULT, DSED, CCLIST, AUTORM, BDBASE, BASE, ARRAYS, CODE, FILES, BACKUP, BACKUP PUNCH, BACKUP PRINTER, BACKUP PUNCH PRINTER

SYSTEM TASK ATTRIBUTE ASSIGNMENT STATEMENT

FUNCTION:

THE SYSTEM TASK ATTRIBUTE ASSIGNMENT STATEMENT IS USED TO SET THE VALUE OF A SYSTEM TASK ATTRIBUTE FOR A TASK.

GENERAL FORMAT:

<SYSTEM TASK ATTRIBUTE>(<TASK DESIGNATOR>)::= <SCALAR EXPRESSION>

GENERAL RULES:

1. ATTEMPTING TO SET A READ-ONLY ATTRIBUTE WILL GENERATE A SYNTAX ERROR OF LEVEL THREE (3) AT COMPILE-TIME AND A DIAGNOSTIC AT RUN-TIME.
2. MULTIPLE ASSIGNMENTS ARE NOT ALLOWED.

3. IF THE TASK ATTRIBUTE NAME IS SPECIFIED, A PERIOD WILL BE INSERTED AT THE END OF THE STRING, IF NOT PRESENT.

4. IF A TASK ATTRIBUTE HAS MNEMONICS THEY MAY BE USED IF DESIRED BUT MNEMONICS ARE NOT REQUIRED.

SYSTEM TASK ATTRIBUTE REFERENCE

FUNCTION:

THE SYSTEM TASK ATTRIBUTE REFERENCE IS USED TO REFERENCE THE VALUE OF A SYSTEM TASK ATTRIBUTE.

GENERAL FORMAT:

<SYSTEM TASK ATTRIBUTE> (<TASK DESIGNATOR>)

GENERAL RULES:

ATTEMPTING TO READ A WRITE-ONLY ATTRIBUTE WILL GENERATE A SYNTAX ERROR OF LEVEL THREE (3) AT COMPILE-TIME AND A DIAGNOSTIC AT RUN-TIME.

D0654 PLI - NEW XREF FORMAT - 12-28-73

THE XREF LISTING HAS BEEN CHANGED IN THE FOLLOWING WAYS:

1. THE IDENTIFIERS ARE LISTED ALPHABETICALLY AND ARE NO LONGER DIVIDED BY BLOCKS.

2. THE FIRST LINE OF EACH ENTRY CONTAINS:

   IDENTIFIER NAME <TYPE> LEX LEVEL

   IF THE IDENTIFIER IS PART OF A STRUCTURE, THE FULLY QUALIFIED NAME IS ADDED TO THE FIRST LINE, FOR EXAMPLE

```
DCL 1 A,
      2 B CHAR(10);
```

   THE XREF ENTRY FOR B WOULD BE:

D0654 PLI - NEW XREF FORMAT - 12-28-73

B <CHAR> AT LEX LEVEL N A.B

3. THE STATEMENT NUMBERS WHERE THE IDENTIFIER HAS BEEN REFERENCED ARE NOW IN THE FORM:

   STATEMENT NUMBER:CARD NUMBER.

D0655 PLI - SIMPLE DEFINE IMPLEMENTATION - 01-06-74

THIS PATCH REPRESENTS THE INITIAL IMPLEMENTATION OF THE SIMPLE DEFINE OPTION OF THE DEFINED ATTRIBUTE. PLEASE REFER TO THE PL/I LANGUAGE MANUAL FOR A COMPLETE EXPLANATION OF THIS FEATURE.

D0656 PLI - FILE ATTRIBUTE ASSIGN AND REF - 01-06-74

IN PL/I D0504 UNDER GENERAL RULES FOR ASSIGNMENT, RULE TWO SHOULD BE CHANGED TO READ:

2. ATTEMPTING TO SET A READ-ONLY ATTRIBUTE WILL GENERATE A SYNTAX ERROR OF LEVEL THREE (3) AT COMPILE-TIME AND A DIAGNOSTIC AT RUN-TIME.

UNDER GENERAL RULES FOR REFERENCE, RULE TWO SHOULD BE CHANGED TO READ:

2. ATTEMPTING TO READ A WRITE-ONLY ATTRIBUTE WILL GENERATE A SYNTAX ERROR OF LEVEL THREE (3) AT COMPILE-TIME AND A DIAGNOSTIC AT RUN-TIME.

D0673 PLI - GENERIC ATTRIBUTE - 12-06-74

THE GENERIC ATTRIBUTE SPECIFIES THAT THE IDENTIFIER HAS BEEN DECLARED AS A GENERIC ENTRY NAME. A REFERENCE TO THE IDENTIFIER REFERS TO THE ENTRY EXPRESSION WHOSE ARGUMENT LIST MATCHES THE PARAMETER LIST IN THE GENERIC REFERENCE.

GENERAL FORMAT:

```
<GENERIC-ATTRIBUTE>::= GENERIC (<GENERIC-ELEMENT>
                                [,<GENERIC-ELEMENT>]...)
<GENERIC-ELEMENT>::= <ENTRY-EXPRESSION> WHEN
                          (<GENERIC-ATTRIBUTE-LIST>)
<GENERIC-ATTRIBUTE-LIST>::= EMPTY / ALIGNED / AREA /
      BINARY [(PRECISION)] / BIT / CHARACTER / DECIMAL / ENTRY
      [(<DESCRIPTION LIST>)] / FILE / FIXED [(PRECISION)] /
      FLOAT [(PRECISION)] / FORMAT / LABEL / OFFSET / PICTURE
      "PICTURE" / POINTER / PRECISION (PRECISION) / REAL
      [(PRECISION)] / RETURNS [(<ATTRIBUTE-LIST>)] / UNALIGNED
<DESCRIPTION-LIST>::= <PARAMETER-ATTRIBUTE-LIST>
                      [,<PARAMETER-ATTRIBUTE-LIST>]...
```

NOTE: THE DOUBLE QUOTE HAS BEEN USED FOR THE SINGLE QUOTE IN THE ABOVE.

GENERAL RULES:

1. THE ONLY ATTRIBUTE WHICH MAY APPEAR WITH GENERIC IS INTERNAL.
2. THE SELECTED ENTRY-EXPRESSION FOR A GIVEN GENERIC-REFERENCE IS THE REFERENCE PART OF ONE OF THE GENERIC ELEMENTS GIVEN IN THE DECLARATION OF THE GENERIC NAME.
3. THE ACTUAL SELECTION DEPENDS ON THE NUMBER OF ARGUMENTS, THEIR ATTRIBUTES, AND THE ORDER IN WHICH THE GENERIC ELEMENTS ARE WRITTEN.
4. FOR A GENERIC REFERENCE WITH NO ARGUMENTS, THE FIRST GENERIC-ELEMENT WITH AN EMPTY GENERIC ATTRIBUTE LIST WILL BE SELECTED.
5. ATTRIBUTES NOT SPECIFIED IN THE GENERIC ATTRIBUTE LIST WILL BE SUPPLIED ACCORDING TO THE DEFAULT RULES.
6. SELECTION OF AN ENTRY EXPRESSION ACCORDING TO THE ARGUMENT ATTRIBUTES REQUIRES THAT ALL THE ATTRIBUTES SPECIFIED FOR A GENERIC ELEMENT BE CONTAINED IN THE ATTRIBUTES OF THE ARGUMENTS.

REMOTE JOB ENTRY (RJE)

D0559 RJE - RJE - WFL INTERFACE - 10-28-73

THE RJE MCS HAS BEEN REWRITTEN FOR THE MARK 2.6 RELEASE IN ORDER TO PROVIDE THE USER OF THE MCS ACCESS TO THE FACILITIES OF THE WORK FLOW MANAGEMENT SYSTEM. THIS REVISION ENTAILED SUBSTANTIAL CHANGES TO THE SOFTWARE REQUIRMENTS OF RJE AND USER INTERFACE.

THIS RJE IMPLEMENTATION REQUIRED ALTERATION OF THE DCALGOL COMPILER AND THE MCP, BUT DID NOT INVOLVE THE DC1000 SOFTWARE. THEREFORE, THIS VERSION OF RJE WILL NEITHER COMPILE WHEN A PRE-2.6 DCALGOL COMPILER IS USED NOR FUNCTION IN A PRE-2.6 MCP ENVIRONMENT. THE MCS DOES NOT, HOWEVER, REQUIRE ANY DC1000 OPERATING SYSTEM OTHER THAN THAT EMPLOYED PRIOR TO THE 2.6 RELEASE.

THE ABILITY OF THE RJE USER TO EMPLOY WORK FLOW MANAGEMENT INPUT MESSAGES AND WORK FLOW LANGUAGE JOB DECKS HAS BEEN GREATLY INCREASED. THE MCS HAS BECOME AN INTERFACE PROGRAM BETWEEN THE RJE USER ENTERING SYSTEM INPUT MESSAGES AT THE REMOTE SUPERVISORY CONSOLE (RSC) AND THE CONTROLLER WHICH SERVICES HIS REQUESTS. THE RSC HAS THUS BECOME A CENTRAL CONSOLE WITH RESTRICTED CAPABILITIES; THE RJE USER MAY EMPLOY ANY SYSTEM INPUT MESSAGE FREED FOR RJE USE IN THE INPUT TO SCTABLEGEN AS WELL AS WORK FLOW LANGUAGE STATEMENTS. ANY SYSTEM INPUT MESSAGES DIRECTED AT THE RJE MCS MUST NOW BE PRECEDED BY THE CHARACTER "*". THESE SYSTEM INPUT MESSAGES INCLUDE THOSE INVOLVING THE AUTOBACKUP ROUTINE OF RJE (I.E., DS, QT, FM, AND PB) AND THE RUN-TIME OPTIONS OF RJE (I.E., SO, TO, RO, SF, TF, AND PH), ETC. THE FORMAT AND EFFECT OF CENTRAL CONSOLE SM COMMANDS DIRECTED AT RJE HAVE NOT BEEN CHANGED.

ANY WORK FLOW LANGUAGE JOB DECK MAY NOW BE READ BY THE REMOTE CARD READER. RJE PASSES THESE DECKS TO A COPY OF THE WORK FLOW LANGUAGE COMPILER WHICH IT PROCESSES TO SERVICE THE TERMINAL. THE WORK FLOW LANGUAGE COMPILER HANDLES ALL DECK PARSING AS WELL AS JOB

D0559 RJE - RJE - WFL INTERFACE - 10-28-73

INITIATION, REPLACING THE FORMER ACTIVITY OF RJE IN THESE AREAS. (SEE THE DOCUMENTATION OF WORK FLOW LANGUAGE IN THE "WORK FLOW MANAGEMENT USERS GUIDE", 5000714.) THE USERCODE OF THE RJE USER IS APPENDED TO THESE DECKS IF THE USER WAS REQUIRED TO LOG ON. MIX-RELATED RSC SYSTEM INPUT MESSAGES MAY APPLY ONLY TO JOBS INITIATED BY SUCH DECKS OR TO JOBS INITIATED IN RESPONSE TO WORK FLOW LANGUAGE STATEMENTS ENTERED AT THE RSC.

THE PRACTICE OF RETAINING CARD DECKS IN THE RJELINKED FILE HAS BEEN DISCONTINUED.

THE "RJEBD" AND "RJESAVEBD" BACKUP DIRECTORIES HAVE BEEN DISCARDED. ALL PRINTER BACKUP FILES ARE NOW CONTAINED IN THE "REMLP<RJE MCS NUMBER>" DIRECTORY; PUNCH BACKUP FILES ARE SIMILARLY CONTAINED IN THE "REMCP<RJE MCS NUMBER>" DIRECTORY. THE TERMINAL LSN IS NO LONGER CONTAINED IN THE BACKUP FILE TITLE.

THE 2.6 VERSION OF THE RJE MCS IS DOCUMENTED IN DETAIL IN THE REVISED "REMOTE JOB ENTRY SYSTEM" INFORMATION MANUAL, 5000300.

D0723 RJE - FILESENTRY - 03-03-74

2.6 RJE NO LONGER USES THE "FILESENTRY" TASK ATTRIBUTE. THIS FEATURE WILL NOT BE MAINTAINED IN FUTURE SYSTEM RELEASES.

SCR

D0636 SCR - FLAG UNIT-DEFAULT BUFFERS - 11-18-73

THE INTERNAL NAME FOR DEFAULT UNITS OR BUFFERS HAS BEEN CHANGED FROM "UNIT" OR "BUFF" TO "*UNIT*" OR "*BUFF*". THIS WILL IMPROVE THE READABILITY OF ERROR LISTINGS THAT AUTOMATICALLY FORCE THE PRINTOUT OF THE DEFAULT NAMES.

D0637 SCR - LINKING IO QUEUE REQUESTS - 11-18-73

WHEN A GIVEN UNIT IS REFERENCED MORE THAN ONCE IN A PARALLEL I/O REQUEST, THE SECOND AND ALL SUBSEQUENT I/O REQUESTS ARE LINKED TO EACH OTHER BEFORE THEY ARE GIVEN TO THE HARDCORE I/O SYSTEM. VIA THIS TECHNIQUE, THE I/O INITIATION OF THE TWO THRU "N" ITEM OF THE SPECIFIED UNIT"S I/O QUEUE IS HANDLED BY THE "HARDWARE INTERRUPT" PROCEDURE. THEREFORE THE TIME DELAY BETWEEN SUCCESSIVE I/O TO THE SAME UNIT WILL BE AS SMALL AS IS POSSIBLE WITHIN THE KERNEL I/O SYSTEM. UNLESS THE SYSTEM HAS ANOTHER UNIT WAITING FOR A PATH TO THE SPECIFIC MINTERM GROUP, AN I/O FINISH ON A GIVEN I/O OPERATION WILL BE IMMEDIATELY FOLLOWED BY AN I/O START-UP OF THE NEXT I/O REQUEST IN THE QUEUE OF THE SAME UNIT. (THIS MAY NOT HAPPEN, HOWEVER, IF THE FIRST I/O RETURNED AN ERROR IN THE RESULT DESCRIPTOR.) IT IS FELT THAT UNDER THIS TECHNIQUE, THE SYSTEM COULD ISSUE I/O TO DISK WITH ONLY A ONE SEGMENT LATENCY REQUIRED, AND THEREBY KEEP THE HEAD-PER-TRACK DISK TRANSFER RATE AT A NEAR MAXIMUM FOR A SPECIFIED NUMBER OF PARALLEL I/O OPERATIONS WHOSE LIMIT IS CURRENTLY SET AT 10. THIS TECHNIQUE ALSO INSURES THAT NO OTHER I/O OPERATION IS QUEUED IN BETWEEN THOSE REQUESTED BY MAT TO THE PARTICULAR UNIT.

D0638 SCR - BUFFER INFO PRIMARY - 11-18-73

THE MAT INTERPRETER WILL PRODUCE A SYNTAX ERROR IF ANY TYPE OF <PRIMARY> OTHER THAN <SIMPLE VARIABLE> OR <NUMBER> IS USED AS THE CHARACTER OFFSET OR <FOR PART> WITHIN THE <BUFFER INFORMATION> PRIMARY.

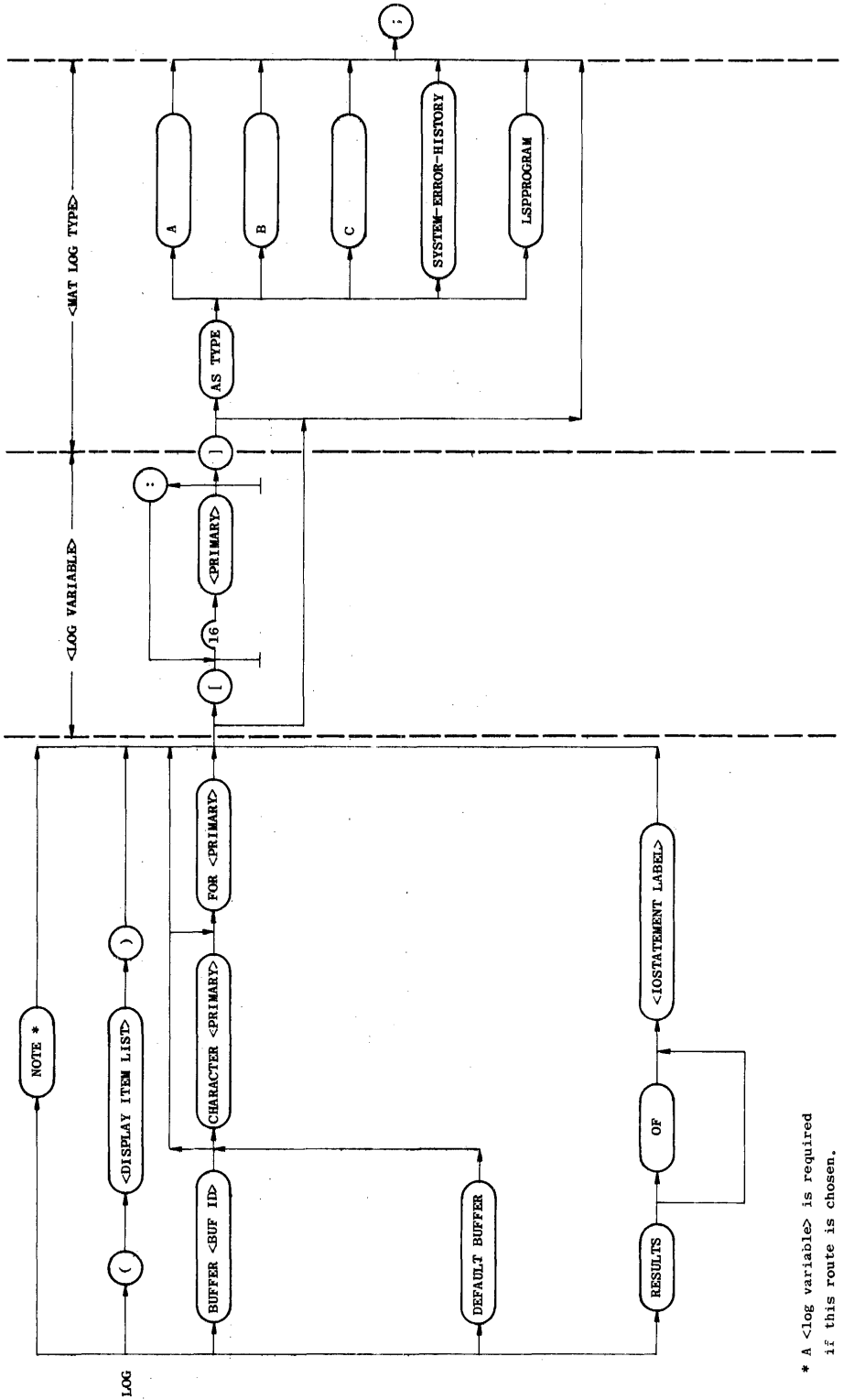
D0639 SCR - LOGGING CHANGES - 12-06-73

THE FOLLOWING FEATURES HAVE BEEN ADDED TO THE MAINTENANCE AND TESTING LANGUAGE FOR LOGGING CAPABILITIES:

1. ABILITY TO WRITE RECORDS INTO THE SYSTEM SUMLOG UNDER THE MAJOR LOGTYPE OF "MAINTMAJOR" AND THE MINOR LOGTYPE OF (FIVE) <MATLOGTYPE>.

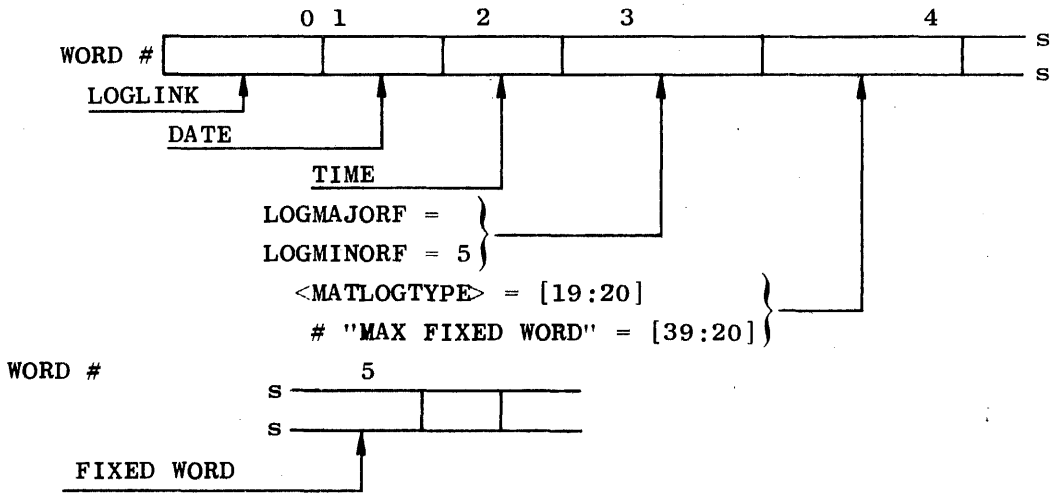
   A. THE SYNTAX FOR THE LOG STATEMENT IS AS FOLLOWS:

LOGGING SYNTAX

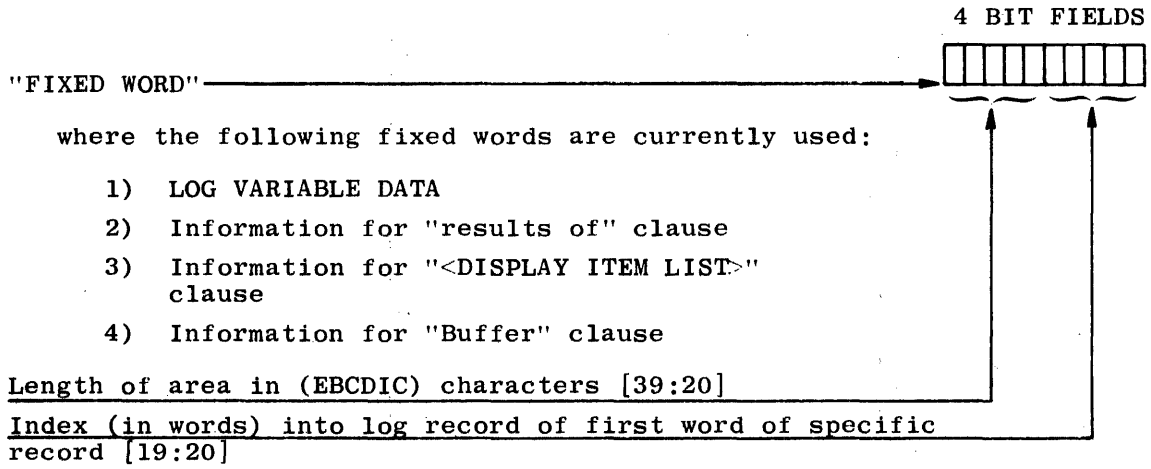


* A <log variable> is required if this route is chosen.

B. THE FORMAT OF THE LOG ENTRY IS AS FOLLOWS: (A MAXIMUM OF 60 WORDS IS ALLOWED PER REQUEST.)



BEGINNING WITH WORD #5 AND FOR THE NUMBER OF WORDS SPECIFIED BY WORD 4.[39:20] THE FOLLOWING WORD FORMAT IS APPLIED:



A FIXED WORD IS PLACED RELATIVE TO WORD FOUR BY ITS TYPE: I.E., "LOG VARIABLE DATA" POINTER IS IN WORD FIVE; "RESULTS OF CLAUSE" POINTER IS IN WORD SIX, ETC.

IF A FIXED WORD TYPE IS NOT AVAILABLE THEN THE APPROPRIATE WORD (RELATIVE TO WORD FOUR (4)) WILL CONTAIN ZEROS.

(WORD #4).[39:20] INDICATES THE MAXIMUM FIXED WORDS AVAILABLE.

THE FOLLOWING FORMATS APPLY TO THE AREAS POINTED AT BY A "FIXED WORD".

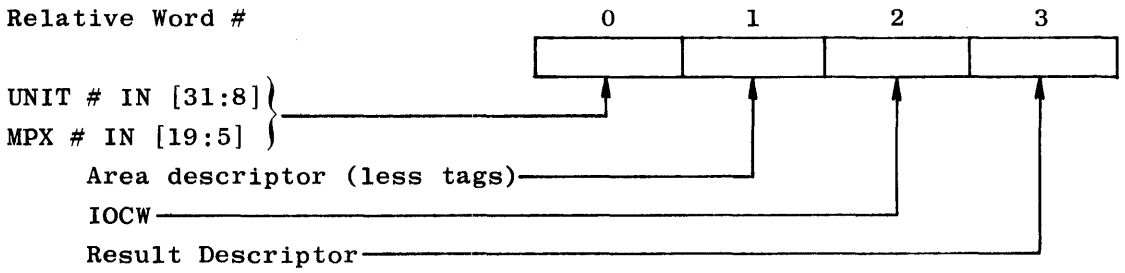
"FIXED WORD" =1: <LOG VARIABLE>

ONE WORD IS USED FOR EACH <PRIMARY> ENTERED, I.E., IF SYNTAX OF [1:2] IS USED, THEN TWO WORDS CONTAINING A ONE, TWO, (1, 2,) RESPECTIVELY, WILL BE PLACED IN THIS AREA.

FOR EXAMPLE, THE "FIXED WORD" POINTING AT THIS ENTRY WOULD CONTAIN A 12 IN [39:20] INDICATING 12 EBCDIC CHARACTERS (OR TWO WORDS) WERE POINTED AT BY THE ENTRY.

"FIXED WORD" =2: "RESULTS OF" CLAUSE

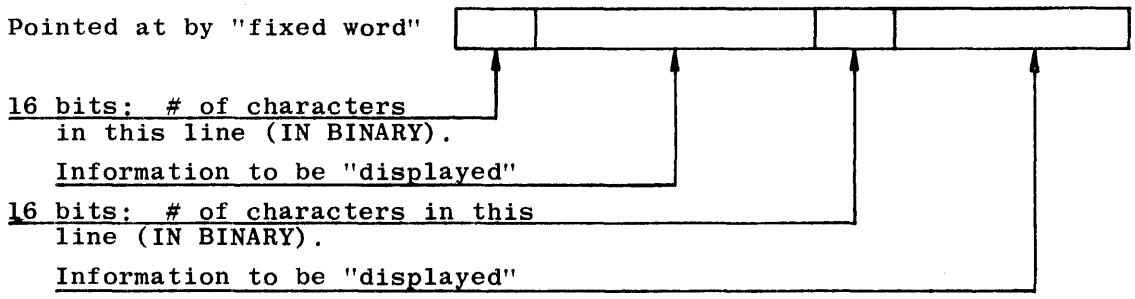
THE "FIXED WORD" FIELD OF [39:20] WILL CONTAIN A 24 (OR FOUR WORDS) AND [19:20] WILL POINT AT THE FOLLOWING:



"FIXED WORD" =3: <DISPLAY ITEM LIST>

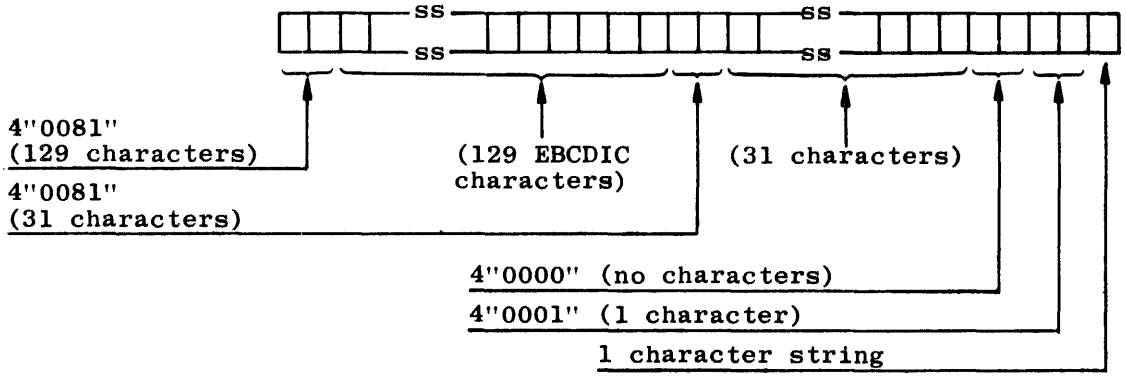
THE "FIXED WORD" FIELD OF [39:20] WILL CONTAIN THE TOTAL LENGTH OF AN EBCDIC STRING (IN CHARACTERS) WHOSE BEGINNING IS LOCATED BY THE WORD INDEX FOUND IN THE

"FIXED WORD" FIELD [19:20]. SINCE THE <DISPLAY ITEM LIST> MAY REPRESENT MORE THAN A PHYSICAL LINE, THE FOLLOWING FORMAT IS EMBEDDED WITHIN THIS TYPE:



NOTE: ANY OF THE 16 BIT FIELDS MAY INDICATE A LENGTH OF ZERO, IMPLYING THAT THE ORIGINAL <DISPLAY ITEM LIST> YIELDED AN <EMPTY> LINE. THE ANALYZER OF THIS RECORD MUST LOOP UNTIL IT HAS EXHAUSTED THE LENGTH OF THE RECORD AS INDICATED BY THE "FIXED WORD" FIELD OF [39:20]. THE [39:20] FIELD ALSO INCLUDES THE TWO CHARACTERS USED TO INDICATE THE LENGTH OF A PARTICULAR "DISPLAY" LINE.

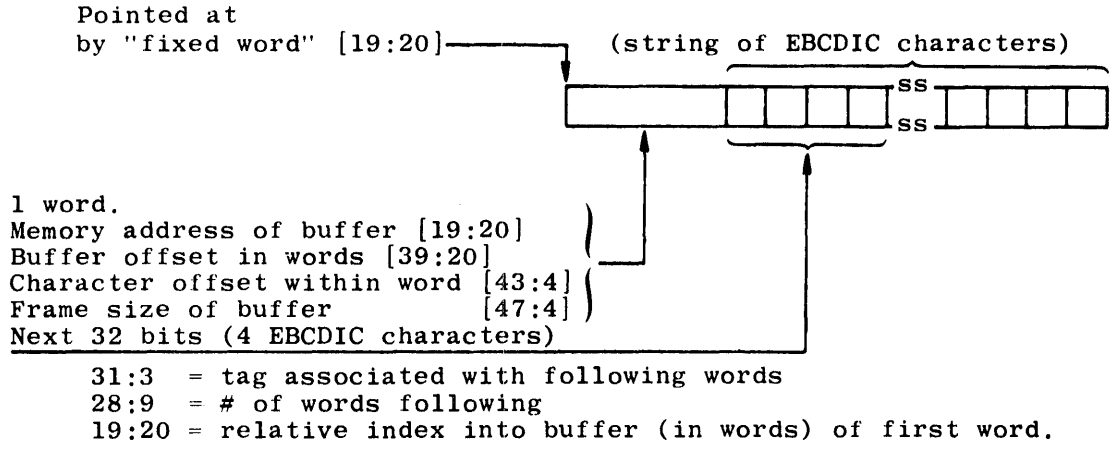
CONSIDER THE FOLLOWING EXAMPLE:



"FIXED WORD" [39:20] WOULD CONTAIN A (129+2) + (31+2) + (0+2) + (1+2), WHICH EQUALS 161+8 OR 169, AS NOTED IN THE ABOVE EXAMPLE.

"FIXED WORD" =4: BUFFER CLAUSE

THE "FIXED WORD" FIELD OF [39:20] WILL CONTAIN THE TOTAL LENGTH (IN CHARACTERS) OF THE BUFFER INFORMATION POINTED AT BY THE FIELD [19:20] OF THE FIXED WORD. IN ORDER TO COMPRESS A BUFFER DUMP INTO THE SMALLEST POSSIBLE AREA, THE FOLLOWING FORMAT HAS BEEN APPLIED:



VIA THIS TECHNIQUE DUPLICATE WORDS WILL NOT APPEAR IN THE LOG INFORMATION. THE NUMBER OF DUPLICATE WORDS MAY BE DETERMINED BY SUBTRACTING THE FIELD OF [19:20] POINTED AT BY [28:9] OF THIS 24-BIT WORD FROM ([19:20] + [28:9] OF THIS 24-BIT WORD).

NOTE THAT THE FIRST WORD (IOCW) IS NOT LOGGED.

D0640 SCR - SUBROUTINE PARAMETERS - 11-18-73

THE ABILITY TO PASS PARAMETERS TO A SUBROUTINE HAS BEEN IMPLEMENTED WITH THE FOLLOWING FEATURES:

1. A MAXIMUM OF EIGHT PARAMETERS MAY BE PASSED; ALL ARE PASSED BY VALUE.

2. SUBROUTINE DECLARATION HAS BEEN EXPANDED TO ALLOW PASSING OF THE PARAMETERS AS FOLLOWS: SUBROUTINE SUB(A,B,C,D).

3. IOSTATEMENTS AND VARIABLES MAY BE PASSED. IF ANY TYPE OTHER THAN A VARIABLE IS PASSED, THEN A SPECIFICATION MUST BE PROVIDED IN THE SUBROUTINE READING AS FOLLOWS:

   IOSTATEMENT B.

4. CALL STATEMENT HAS BEEN MODIFIED TO ALLOW PASSING THE PARAMETERS TO THE SUBROUTINE, FOR EXAMPLE:

   CALL SUB (1,S,S(MPXNO),J).

5. IDENTIFIERS DECLARED AS PARAMETERS TO A SUBROUTINE MUST BE UNIQUE; I.E., THERE MAY NOT BE A PARAMETER "A" AND A GLOBAL OR LOCAL IDENTIFIER "A".

D0641 SCR - DISPLAY-SPOUT CHANGES - 11-18-73

DISPLAY/SPOUT STATEMENTS HAVE BEEN IMPROVED AS FOLLOWS:

1. A SIX BIT BUFFER MAY NOW BE DISPLAYED.

2. THERE ARE NO LENGTH CHECKS PERFORMED AT COMPILE-TIME. IF AT RUN-TIME, THE LENGTH OF THE LINE TO BE DISPLAYED OR SPOUTED EXCEEDS A GIVEN LINE LENGTH (120 FOR A PRINTER, 40 FOR A SPO) A WARNING MESSAGE IS GENERATED AND THE REMAINING DATA IS NOT TRANSMITTED.

3. ABILITY TO CAUSE SPACING WITHIN A GIVEN DISPLAY STATEMENT THROUGH USE OF A "/" OR "/ FOR <NUMBER>" HAS BEEN IMPLEMENTED. THE USE OF THIS WITHIN A <LOG STATEMENT> IS INHIBITED.

4. ABILITY TO DUMP A <PRIMARY> "IN EBCDIC" OR "IN BCL" AS WELL AS "IN HEX" AND "IN OCTAL" HAS BEEN IMPLEMENTED. (THE DEFAULT IS DECIMAL.)

5. ABILITY TO DISPLAY OR SPOUT A DEFAULT BUFFER BY SYNTAX OF

D0641 SCR - DISPLAY-SPOUT CHANGES - 11-18-73

"DEFAULT BUFFER" HAS BEEN IMPLEMENTED.

6. ABILITY TO DISPLAY OR SPOUT <BUFFER INFORMATION> BY ENCLOSING ANY <PRIMARY> IN PARENTHESES HAS BEEN IMPLEMENTED.

7. THE LENGTH OF ITEMS DISPLAYED "IN OCTAL" AND "IN HEX" IS REDUCED TO THE SMALLEST NUMBER OF CHARACTERS REQUIRED. (THIS MAY BE OVERRIDDEN BY THE USE OF THE <FOR PART>.)

D0639 SCR - REMOVE LOG CODES FOR LSP - 11-18-73

DUE TO THE GENERALITY OF THE NEW LOG SYNTAX, IT IS FELT THAT ONE LOG TYPE SHOULD GENERATE A SYNTAX ERROR IF NOT USED WITH A CORRESPONDING LOG TYPE. IN THIS WAY ACCIDENTAL USAGE OF THIS TYPE MAY BE PROVIDED, YET OVERRIDES ARE PERMITTED BY FIELD PERSONNEL IF NECESSARY.

THIS PATCH IMPLEMENTS A NEW DOLLAR CARD

$SET LSPPROGRAM

WHICH IS USED IN CONJUNCTION WITH LOG TYPE "LSPPROGRAM".

D0644 SCR - CHAR OFFSET IN SPOUT-DISPLAY - 12-06-73

THIS PATCH ALLOWS SPECIFYING A CHARACTER OFFSET AND <FOR PART> AS PART OF THE BUFFER PORTION OF A DISPLAY OR SPOUT STATEMENT, FOR EXAMPLE:

DISPLAY(BUFFER CHAR I FOR J)

NOTE THAT IF THE ABOVE WAS TO BE THE <PRIMARY><BUFFER INFORMATION> THEN IT MUST BE ENCLOSED IN PARENTHESES.

D0645 SCR - LOADING BX383 FIRMWARE - 12-06-73

THIS PATCH PROVIDES A METHOD FOR LOADING BX383 FIRMWARE. HOWEVER, HARDWARE INTERRUPTION IS REQUIRED, AND AT THE TIME OF THE LOAD, NO OTHER PACKS WITHIN THE SPECIFIED MINTERM GROUP MAY BE IN USE BY THE SYSTEM. THE SYNTAX IS AS FOLLOWS:

```
LOAD <UNIT SPECIFIER> FILEADDR 0 ON ERROR (MAT MASK=
                        HEX"03000");
   (BX383 FIRMWARE DECK)
ENDLOAD;
```

A <MPX ROUTE> MUST BE SPECIFIED WITHIN <UNIT SPECIFIED>.

D0646 SCR - SET <BUFFER INFO> =... - 01-20-74

THIS CHANGE PROVIDES SCR WITH THE ABILITY TO STORE INFORMATION ORIGINATING IN A PRIMARY INTO A SPECIFIED CHARACTER OF A BUFFER. THE FOLLOWING SYNTAX HAS BEEN PROVIDED:

```
SET <BUFFER INFO> = ...
```

WHICH MAY BE EXPANDED TO:

```
SET BUFFER X CHAR 5 FOR 2 = HEX"FFFF";
```

THE BUFFER CHARACTER SIZE DETERMINES THE SIZE OF THE CHARACTER WITHIN THE PRIMARY. THE <FOR PART> MAY NOT EXCEED FOUR CHARACTERS.

D0647 SCR - REWIND AND WAIT STMT - 01-20-74

THE REWIND AND WAIT STATEMENT IS TERMINATED BY AN I/O TIMEOUT IF AN ENTIRE REEL IS REMOVED. THIS PATCH ALLOWS TWO MINUTES TO ELAPSE BEFORE THE I/O TIMEOUT IS INVOKED. THIS TWO MINUTES MAY BE OVERRIDDEN BY A SET IOTIMEOUT INSTRUCTION.

D0648 SCR - VARIABLE RUN PARAMETERS - 01-20-74

THE ABILITY TO PASS PARAMETERS TO A SCR PROGRAM WAS REQUIRED FOR THE COMPLETION OF THE VARIOUS MAT PROJECTS. THE FOLLOWING SYNTAX HAS BEEN IMPLEMENTED:

RUN <PROG NAME> (<IDENT1>=<NUMBER>,ETC)

AND A MAT STATEMENT:

RUNPARMS (<VARIABLE IDENTIFIER> [<IDENT1>],ETC)

THE <IDENT1> VALUE MAY BE USED TO REPLACE THE NAME THAT THE USER MUST ENTER THROUGH THE RUN STATEMENT. A MAXIMUM OF 47 PARAMETERS MAY BE PASSED TO A MAT PROGRAM VIA THIS TECHNIQUE. IF THE <RUNPARMS STATEMENT> DOES NOT CONTAIN THE OPTIONAL "[IDENT1]" CLAUSE THE <RUN STATEMENT> <IDENT1> MUST REFER TO THE <VARIABLE IDENTIFIER> REFERENCED IN THE <RUNPARMS STATEMENT>.

SORT

D0623 SORT - SORT LOOPING ON USER OUTPUT - 12-28-73

THIS CHANGE WILL CAUSE THE SORT TO REQUIRE OPERATOR INTERVENTION WHEN IRRECOVERABLE I/O ERRORS OCCUR ON A NON-DISK USER OUTPUT FILE (USUALLY TAPE). PREVIOUSLY THE SORT WOULD RETRY AUTOMATICALLY AND SOMETIMES GET INTO A LOOP BY REUSING THE BAD OUTPUT TAPE. THE PROBLEM INVOLVES DISK SORT WITH OR WITHOUT RESTART. THE SORT WILL NOW DISPLAY SORT ERROR 13 AND THE OPERATOR MUST OK TO CONTINUE OR DS TO STOP.

SOURCENDL

D0624 SOURCENDL - PAGE AND VERSION - 12-28-73

THE $OPTION VERSION HAS BEEN ADDED TO THE SOURCENDL SYMBOLIC, AS WELL AS $PAGE PRIOR TO EACH NDL SECTION.

D0625 SOURCENDL - TABS FOR MODEL 2741 TERMINAL - 12-28-73

THIS CHANGE IMPLEMENTS AUTOMATIC TABULATION FOR MODEL 2741 TERMINALS. FOR INPUT, THE TAB CHARACTER CAUSES BLANKS TO BE INSERTED IN THE MESSAGE UP TO THE NEXT TAB STOP. ON OUTPUT, TAB CHARACTERS WILL BE TRANSMITTED IN PLACE OF BLANKS WHERE APPROPRIATE. TABS ARE ASSUMED TO BE SET EVERY FIVE CHARACTERS. THIS SIGNIFICANTLY INCREASES THE SPEED OF THE TERMINAL.

D0626 SOURCENDL - B9352 ETX IN MESSAGE - 12-28-73

IF AN ETX CHARACTER IS FOUND IN A MESSAGE BEING SENT TO A B9352 TERMINAL, ANY REMAINING TEXT IN THIS MESSAGE WILL BE IGNORED; ETX AND BCC ARE SUBSEQUENTLY TRANSMITTED.

WORK FLOW LANGUAGE (WFL)

D0578 WFL - UNARY PLUS OPERATOR - 11-18-73

THIS CHANGE ALLOWS THE USE OF THE UNARY PLUS OPERATOR. FOR EXAMPLE:

```
I:= +1;
```

D0677 WFL - DECK STATEMENT - 01-27-74

THE DECK STATEMENT ALLOWS A USER TO CREATE A DATA FILE ON DISK CONSISTING OF EBCDIC OR BCL CARD IMAGES. THE DECK STATEMENT MAY NOT BE PART OF A JOB AND MUST BE THE FIRST CARD OF A JOB. A DATA STATEMENT MUST FOLLOW THE DECK STATEMENT AND MUST SPECIFY A FILE NAME TO BE USED AS THE NAME OF THE NEW FILE. THE NEW FILE HAS THE FOLLOWING FILE ATTRIBUTES:

```
KIND=DISK
AREASIZE=100
BLOCKSIZE=450
MAXRECSIZE=15 (EBCDIC)
           10 (BCL)
INTMODE=4 (EBCDIC)
        3 (BCL)
```

EXAMPLE:

```
<I> DECK
<I> EBCDIC        GLOBAL/DECK
      .
      .
      .
      .
```

<I> END

D0613 WFL - NEW TASK ATTRIBUTES - 02-03-74

TWO NEW TASK ATTRIBUTES, DESTNAME (SYNONYM BACKUPDESTINATION) AND MAXWAIT HAVE BEEN ADDED TO WORK FLOW LANGUAGE. PLEASE REFER TO THE DESCRIPTION OF NEW TASK ATTRIBUTES IN MCP D0581.

DOCUMENTS AFFECTED

DOCUMENT	SYSTEM NOTE	MARKETING NO.	MARKETING DATE
ALGOL COMPILER	D0551	5000136	06-72
ALGOL COMPILER	D0592	5000136	06-72
ALGOL COMPILER	D0659	5000136	06-72
ALGOL COMPILER	D0696	5000136	06-72
ALGOL LANGUAGE	D0583	5000128	06-72
ALGOL LANGUAGE	D0592	5000128	06-72
ALGOL LANGUAGE	D0612	5000128	06-72
ALGOL LANGUAGE	D0617	5000128	06-73
ALGOL LANGUAGE	D0643	5000128	06-72
ALGOL LANGUAGE	D0658	5000128	06-72
ALGOL LANGUAGE	D0682	5000128	06-72
ALGOL LANGUAGE	D0697	5000128	06-72
ALGOL LANGUAGE	D0707	5000128	06-72
APL INSTALLATION	D0720	5000805	03-74
BASIC LANGUAGE	D0584	5000383	07-71
BASIC LANGUAGE	D0585	5000383	07-71
BASIC LANGUAGE	D0586	5000383	07-71
BASIC LANGUAGE	D0587	5000383	07-71
BASIC LANGUAGE	D0588	5000383	07-71
BASIC LANGUAGE	D0589	5000383	07-71
BASIC LANGUAGE	D0596	5000383	07-71
BASIC LANGUAGE	D0683	5000383	07-71
COBOL REFERENCE	D0617	5000656	02-73
COBOL REFERENCE	D0660	5000656	01-74
COBOL REFERENCE	D0662	5000656	01-74
COBOL REFERENCE	D0663	5000656	01-74
COBOL REFERENCE	D0664	5000656	01-74
COBOL REFERENCE	D0684	5000656	01-74
COBOL REFERENCE	D0685	5000656	01-74
COBOL REFERENCE	D0686	5000656	01-74

DOCUMENTS AFFECTED

DOCUMENT	SYSTEM NOTE	MARKETING NO.	MARKETING DATE
COBOL REFERENCE	D0687	5000656	01-74
COBOL REFERENCE	D0688	5000656	01-74
COBOL REFERENCE	D0689	5000656	01-74
COBOL REFERENCE	D0690	5000656	01-74
COBOL REFERENCE	D0691	5000656	01-74
COBOL REFERENCE	D0692	5000656	01-74
COBOL REFERENCE	D0693	5000656	01-74
COBOL REFERENCE	D0698	5000656	01-74
COBOL REFERENCE	D0700	5000656	01-74
COBOL REFERENCE	D0701	5000656	01-74
COBOL REFERENCE	D0702	5000656	01-74
COBOL REFERENCE	D0703	5000656	01-74
COBOL REFERENCE	D0704	5000656	01-74
COBOL REFERENCE	D0705	5000656	01-74
COBOL REFERENCE	D0708	5000656	01-74
COBOL REFERENCE	D0709	5000656	01-74
COBOL REFERENCE	D0710	5000656	01-74
COBOL REFERENCE	D0714	5000656	01-74
COBOL REFERENCE	D0721	5000656	01-74
COBOL REFERENCE	D0722	5000656	01-74
COBOL REFERENCE	D0725	5000656	01-74
D NOTE DOCUMENTATION	D0557	5000763	10-73
D NOTE DOCUMENTATION	D0597	5000763	10-73
DATA MANAGEMENT	D0719	5000235	01-73
DCALGOL REFERENCE	D0563	5000052	06-73
DCALGOL REFERENCE	D0579	5000052	06-73
DCALGOL REFERENCE	D0582	5000052	06-73
DCALGOL REFERENCE	D0598	5000052	06-73
DCALGOL REFERENCE	D0599	5000052	06-73
DCALGOL REFERENCE	D0600	5000052	06-73
DCALGOL REFERENCE	D0601	5000052	06-73
DCALGOL REFERENCE	D0602	5000052	06-73
DCALGOL REFERENCE	D0604	5000052	06-73

DOCUMENTS AFFECTED

DOCUMENT	SYSTEM NOTE	MARKETING NO.	MARKETING DATE
DCALGOL REFERENCE	D0699	5000052	06-73
DUMP ANALYZER	D0546	5000334	11-71
DUMP ANALYZER	D0576	5000334	11-71
DUMP ANALYZER	D0609	5000334	11-71
DUMP ANALYZER	D0610	5000334	11-71
DUMP ANALYZER	D0627	5000334	11-71
ESPOL LANGUAGE	D0590	5000095	06-72
ESPOL LANGUAGE	D0591	5000095	06-72
ESPOL LANGUAGE	D0611	5000095	06-72
ESPOL LANGUAGE	D0666	5000095	06-72
FORTRAN REFERENCE	D0680	5000458	06-72
FORTRAN REFERENCE	D0681	5000458	06-72
FORTRAN REFERENCE	D0694	5000458	06-72
FORTRAN REFERENCE	D0706	5000458	06-72
I-O SUBSYSTEM	D0565	5000185	07-71
I-O SUBSYSTEM	D0574	5000185	07-71
I-O SUBSYSTEM	D0580	5000185	07-71
I-O SUBSYSTEM	D0595	5000185	07-71
NDL	D0560	5000078	08-71
NDL	D0561	5000078	08-71
NDL	D0575	5000078	08-71
NDL	D0605	5000078	08-71
NDL	D0606	5000078	08-71
NDL	D0622	5000078	08-71
NDL	D0667	5000078	08-71
NDL	D0716	5000078	08-71
NDL	D0726	5000078	08-71
ON-LINE MAT LANGUAGE	D0636	5000169	10-73
ON-LINE MAT LANGUAGE	D0637	5000169	10-73
ON-LINE MAT LANGUAGE	D0638	5000169	10-73
ON-LINE MAT LANGUAGE	D0639	5000169	10-73
ON-LINE MAT LANGUAGE	D0639	5000169	10-73
ON-LINE MAT LANGUAGE	D0640	5000169	10-73

DOCUMENTS AFFECTED

DOCUMENT	SYSTEM NOTE	MARKETING NO.	MARKETING DATE
ON-LINE MAT LANGUAGE	D0641	5000169	10-73
ON-LINE MAT LANGUAGE	D0644	5000169	10-73
ON-LINE MAT LANGUAGE	D0645	5000169	10-73
ON-LINE MAT LANGUAGE	D0646	5000169	10-73
ON-LINE MAT LANGUAGE	D0647	5000169	10-73
ON-LINE MAT LANGUAGE	D0648	5000169	10-73
PL-I LANGUAGE	D0649	5000201	10-72
PL-I LANGUAGE	D0650	5000201	10-72
PL-I LANGUAGE	D0651	5000201	10-72
PL-I LANGUAGE	D0652	5000201	10-72
PL-I LANGUAGE	D0653	5000201	10-72
PL-I LANGUAGE	D0654	5000201	10-72
PL-I LANGUAGE	D0655	5000201	10-72
PL-I LANGUAGE	D0656	5000201	10-72
PL-I LANGUAGE	D0673	5000201	10-72
SOFTWARE HANDBOOK	D0543	5000722	07-73
SOFTWARE HANDBOOK	D0545	5000722	07-73
SOFTWARE HANDBOOK	D0548	5000722	07-73
SOFTWARE HANDBOOK	D0551	5000722	07-73
SOFTWARE HANDBOOK	D0553	5000722	07-73
SOFTWARE HANDBOOK	D0554	5000722	07-73
SOFTWARE HANDBOOK	D0558	5000722	07-73
SOFTWARE HANDBOOK	D0562	5000722	07-73
SOFTWARE HANDBOOK	D0565	5000722	07-73
SOFTWARE HANDBOOK	D0566	5000722	07-73
SOFTWARE HANDBOOK	D0573	5000722	07-73
SOFTWARE HANDBOOK	D0574	5000722	07-73
SOFTWARE HANDBOOK	D0577	5000722	07-73
SOFTWARE HANDBOOK	D0581	5000722	07-73
SOFTWARE HANDBOOK	D0593	5000722	07-73
SOFTWARE HANDBOOK	D0613	5000722	07-73
SOFTWARE HANDBOOK	D0615	5000722	07-73
SOFTWARE HANDBOOK	D0617	5000722	07-73

DOCUMENTS AFFECTED

DOCUMENT	SYSTEM NOTE	MARKETING NO.	MARKETING DATE
SOFTWARE HANDBOOK	D0618	5000722	07-73
SOFTWARE HANDBOOK	D0630	5000722	07-73
SOFTWARE HANDBOOK	D0631	5000722	07-73
SOFTWARE HANDBOOK	D0632	5000722	07-73
SOFTWARE HANDBOOK	D0633	5000722	07-73
SOFTWARE HANDBOOK	D0634	5000722	07-73
SOFTWARE HANDBOOK	D0668	5000722	07-73
SOFTWARE HANDBOOK	D0669	5000722	07-73
SOFTWARE HANDBOOK	D0674	5000722	07-73
SOFTWARE HANDBOOK	D0675	5000722	07-73
SOFTWARE HANDBOOK	D0677	5000722	07-73
SOFTWARE HANDBOOK	D0678	5000722	07-73
SOFTWARE HANDBOOK	D0711	5000722	07-73
SOFTWARE HANDBOOK	D0712	5000722	07-73
SOFTWARE HANDBOOK	D0715	5000722	07-73
SOFTWARE HANDBOOK	D0718	5000722	07-73
SORT	D0623	5000144	12-71
SYSTEM MISCELLANEA	D0549	5000367	04-73
SYSTEM MISCELLANEA	D0555	5000367	04-73
SYSTEM MISCELLANEA	D0564	5000367	04-73
SYSTEM MISCELLANEA	D0566	5000367	04-73
SYSTEM MISCELLANEA	D0567	5000367	04-73
SYSTEM MISCELLANEA	D0568	5000367	04-73
SYSTEM MISCELLANEA	D0569	5000367	04-73
SYSTEM MISCELLANEA	D0570	5000367	04-73
SYSTEM MISCELLANEA	D0581	5000367	10-73
SYSTEM MISCELLANEA	D0607	5000367	04-73
SYSTEM MISCELLANEA	D0608	5000367	04-73
SYSTEM MISCELLANEA	D0613	5000367	10-73
SYSTEM MISCELLANEA	D0614	5000367	04-73
SYSTEM MISCELLANEA	D0616	5000367	04-73
SYSTEM MISCELLANEA	D0619	5000367	04-73
SYSTEM MISCELLANEA	D0624	5000367	04-73

DOCUMENTS AFFECTED

DOCUMENT	SYSTEM NOTE	MARKETING NO.	MARKETING DATE
SYSTEM MISCELLANEA	D0625	5000367	04-73
SYSTEM MISCELLANEA	D0626	5000367	04-73
SYSTEM MISCELLANEA	D0665	5000367	04-73
SYSTEM MISCELLANEA	D0668	5000367	04-73
SYSTEM MISCELLANEA	D0669	5000367	04-73
SYSTEM MISCELLANEA	D0670	5000367	04-73
SYSTEM MISCELLANEA	D0671	5000367	04-73
SYSTEM MISCELLANEA	D0672	5000367	04-73
SYSTEM MISCELLANEA	D0675	5000367	10-73
SYSTEM MISCELLANEA	D0676	5000367	04-73
SYSTEM MISCELLANEA	D0713	5000367	04-73
USERDATA-MAKEUSER	D0567	5000797	03-74
USERDATA-MAKEUSER	D0568	5000797	03-74
USERDATA-MAKEUSER	D0569	5000797	03-74
USERDATA-MAKEUSER	D0570	5000797	03-74
WFM REFERENCE	D0571	5000706	04-73
WFM REFERENCE	D0577	5000706	04-73
WFM REFERENCE	D0578	5000706	04-73
WFM REFERENCE	D0593	5000706	04-73
WFM REFERENCE	D0628	5000706	04-73
WFM REFERENCE	D0695	5000706	04-73
WFM USERS GUIDE	D0543	5000714	04-73
WFM USERS GUIDE	D0545	5000714	04-73
WFM USERS GUIDE	D0547	5000714	04-73
WFM USERS GUIDE	D0553	5000714	04-73
WFM USERS GUIDE	D0558	5000714	04-73
WFM USERS GUIDE	D0562	5000714	04-73
WFM USERS GUIDE	D0565	5000714	04-73
WFM USERS GUIDE	D0566	5000714	04-73
WFM USERS GUIDE	D0571	5000714	04-73
WFM USERS GUIDE	D0574	5000714	04-73
WFM USERS GUIDE	D0577	5000714	04-73
WFM USERS GUIDE	D0593	5000714	04-73

DOCUMENTS AFFECTED

DOCUMENT	SYSTEM NOTE	MARKETING NO.	MARKETING DATE
WFM USERS GUIDE	D0613	5000714	04-73
WFM USERS GUIDE	D0621	5000714	04-73
WFM USERS GUIDE	D0677	5000714	04-73
WFM USERS GUIDE	D0711	5000714	04-73
WFM USERS GUIDE	D0717	5000714	04-73

KWIC SUBJECT INDEX

KWIC SUBJECT INDEX

KWIC	SYSTEM NOTE	FUNCTION
ASSIGN AND REF FILE ATTRIBUTE	D0656	PLI
ASSIGN AND REF TASK ATTRIBUTE	D0653	PLI
ATTRIBUTE GENERIC	D0673	PLI
ATTRIBUTE SERIALNO	D0691	COBOL
ATTRIBUTE DIRECT I-O BUFFER	D0617	MCP-I-O
ATTRIBUTE ASSIGN AND REF FILE	D0656	PLI
ATTRIBUTE ASSIGN AND REF TASK	D0653	PLI
ATTRIBUTE REVISIONS FILE	D0565	MCP-I-O
ATTRIBUTES TASK	D0581	MCP
ATTRIBUTES NEW TASK	D0613	WFL
ATTRIBUTES TRANSLATE	D0658	ALGOL
ATTRIBUTES FORTRAN FILE	D0706	FORTRAN
ATTRIBUTES TO DCALGOL RJE	D0699	DCALGOL
AUTO BAD FILE REMOVAL-LOADER	D0545	LOADER
BACKUP FILES ON PACK	D0631	BACKUP
BACKUP TAPES B5700	D0678	BACKUP
BACKUP TO PACK	D0577	MCP
BAD FILE REMOVAL-LOADER AUTO	D0545	LOADER
BAD PARAMETER DETECTION	D0597	LISTDIR
BATCH COMPILING - ALGOL	D0592	ALGOL
BEFORE 2.6 NDL FILES	D0716	MCP-DATACM
BLOCK WRITE	D0693	COBOL
BREAK" STATEMENT "INITIATE	D0605	DCPPROGEN
BUFFER ATTRIBUTE DIRECT I-O	D0617	MCP-I-O
BUFFER INFO PRIMARY	D0638	SCR
BUFFERS FLAG UNIT-DEFAULT	D0636	SCR
BX383 FIRMWARE LOADING	D0645	SCR
BYTE VARIABLE "AI" ADDED AS	D0606	DCPPROGEN
BYTE VARIABLE CRC[0],CRC[1] AS	D0575	NDL
B5700 BACKUP TAPES	D0678	BACKUP
B9352 ETX IN MESSAGE	D0626	SOURCENDL
CALCULATION MEMORY INTEGRAL	D0621	MCP
CAPABILITIES REMOTESPO	D0718	CONTROLLER

KWIC SUBJECT INDEX

KWIC SUBJECT INDEX

KWIC SUBJECT INDEX

KWIC	SYSTEM NOTE	FUNCTION
DETECTION I-O ERROR	D0724	MCP
DETECTION BAD PARAMETER	D0597	LISTDIR
DIAGNOSTIC DCWRITE FUNCTIONS	D0582	MCP-DATACM
DIAGNOSTIC MCS REQUESTS	D0604	DCPPROGEN
DIMENSIONAL ARRAYS TWO	D0685	COBOL
DIRECT I-O BUFFER ATTRIBUTE	D0617	MCP-I-O
DISK FILES PATCH DECKS ON	D0634	PATCH
DISK-PACK ERROR RECOVERY	D0550	MCP
DISPLAY OF STATUS CHANGES	D0711	CONTROLLER
DISPLAY STATEMENT	D0686	COBOL
DISPLAY-SPOUT CHANGES	D0641	SCR
DISPOSTION RECALL; PURGE	D0570	MAKEUSER
DLS FOR DCWRITE ALLOW	D0598	MCP-DATACM
DMS II ALGOL INTERFACE TO	D0583	ALGOL
DOLLAR CARDS	D0551	ALGOL
DOLLAR CARDS	D0722	COBOL
DOLLAR OPTION TIME	D0700	COBOL
DOLLAR OPTION CTMON	D0659	ALGOL
DOLLAR OPTION CHECKPOINT	D0590	ESPOL
DOLLAR OPTION LISTOMITTED	D0696	ALGOL
DOLLAR OPTION "PAGE"	D0584	BASIC
DOLLAR OPTION NOWWARN ADD	D0667	NDL
DOUBLE-TO-SINGLE OPTION	D0694	FORTRAN
DUMPALL INITIAL RELEASE	D0675	DUMPALL
EMPTY LINE DCP RESULTS ON	D0601	MCP-DATACM
ENHANCED DATA MANAGEMENT	D0572	MCP
ENVIR DCWRITE INTERROGATE STA	D0579	MCP-DATACM
ERROR DETECTION I-O	D0724	MCP
ERROR RECOVERY DISK-PACK	D0550	MCP
ESPOL $EXCLUDE TO	D0666	ESPOL
ESPOL LINENUMBER TO	D0611	ESPOL
ESPOL CONSTRUCT, CONVERT NEW	D0591	ESPOL
ETX IN MESSAGE B9352	D0626	SOURCENDL

KWIC SUBJECT INDEX

KWIC	SYSTEM NOTE	FUNCTION
EXAMINE WITH "LOW-VALUE"	D0714	COBOL
EXPRESSIONS ARITHMETIC	D0692	COBOL
EXTERNAL DECKS IN WFL	D0629	MCP
FAILURE DCP CLUSTER	D0554	MCP-DATACM
FILE ATTRIBUTE ASSIGN AND REF	D0656	PLI
FILE ATTRIBUTE REVISIONS	D0565	MCP-I-O
FILE ATTRIBUTES FORTRAN	D0706	FORTRAN
FILE DECLARATIONS	D0651	PLI
FILE DESCRIPTIONS	D0709	COBOL
FILE GENEALOGY	D0580	MCP-I-O
FILE LOADER ONLY LOADS MCPCODE	D0614	LOADER
FILE REMOVAL-LOADER AUTO BAD	D0545	LOADER
FILES UP-TAPE	D0674	BACKUP
FILES IDENTIFY SYSTEM	D0712	LISTFILES
FILES PATCH DECKS ON DISK	D0634	PATCH
FILES SEGMENTATION OF CODE	D0596	BASIC
FILES WRITE STMT FOR REMOTE	D0701	COBOL
FILES BEFORE 2.6 NDL	D0716	MCP-DATACM
FILES FOR IPC PROGRAMS GLOBAL	D0695	MCP-I-O
FILES ON PACK BACKUP	D0631	BACKUP
FILES ON 2.7 READ-ONLY DCPCODE	D0726	NDL
FILES WITH ODD LENGTH RECORDS	D0683	BASIC
FILESENTRY	D0723	RJE
FILLER ITEMS	D0662	COBOL
FIRMWARE LOADING BX383	D0645	SCR
FLAG CONFLICTS IN REPORT #4	D0632	LISTDIR
FLAG MISLEADING RESULTS	D0671	PACKDIR
FLAG UNIT-DEFAULT BUFFERS	D0636	SCR
FOR DCWRITE ALLOW DLS	D0598	MCP-DATACM
FOR IPC PROGRAMS GLOBAL FILES	D0695	MCP-I-O
FOR LSP REMOVE LOG CODES	D0639	SCR
FOR MODEL 2741 TERMINAL TABS	D0625	SOURCENDL
FOR REMOTE FILES WRITE STMT	D0701	COBOL

KWIC SUBJECT INDEX

KWIC SUBJECT INDEX

KWIC		SYSTEM NOTE	FUNCTION
INDEX-NAMES	OWN	D0689	COBOL
INFO PRIMARY	BUFFER	D0638	SCR
INFO> =...	SET <BUFFER	D0646	SCR
INITIAL RELEASE	DUMPALL	D0675	DUMPALL
INPUT WITHOUT "$"	CONTROL	D0713	MAKEUSER
INPUT-OUTPUT CAPABILITY		D0567	MAKEUSER
INSTALLATION INTRINSICS		D0698	COBOL
INTEGRAL CALCULATION	MEMORY	D0621	MCP
INTERFACE	RJE - WFL	D0559	RJE
INTERFACE TO DMS II	ALGOL	D0583	ALGOL
INTERROGATE STA ENVIR	DCWRITE	D0579	MCP-DATACM
INTRINSIC FUNCTIONS		D0708	COBOL
INTRINSICS	INSTALLATION	D0698	COBOL
INTRINSICS	REMOVE AND CHANGE	D0612	ESPOLINTRN
IO QUEUE REQUESTS	LINKING	D0637	SCR
IPC PROGRAMS	GLOBAL FILES FOR	D0695	MCP-I-O
ITEMS	FILLER	D0662	COBOL
ITEMS	USASI OPTION - COMP	D0721	COBOL
JOBS WITH PUNCH OUTPUT		D0628	MCP
KINDLISTS	SYSTEM USE OF	D0595	MCP-I-O
LABEL RECORDS	USER	D0702	COBOL
LENGTH RECORDS	FILES WITH ODD	D0683	BASIC
LIMIT MAXIMUM MCS"S		D0561	NDL
LINAGE-COUNTER		D0705	COBOL
LINE	DCP RESULTS ON EMPTY	D0601	MCP-DATACM
LINE "DEF" FUNCTIONS	SINGLE	D0589	BASIC
LINE COUNT	TOTAL MODS ON	D0609	DUMPANALY
LINENUMBER ADDED TO ALGOL		D0697	ALGOL
LINENUMBER TO ESPOL		D0611	ESPOL
LINKING IO QUEUE REQUESTS		D0637	SCR
LISTOMITTED DOLLAR OPTION		D0696	ALGOL
LITERALS	IMBEDDED QUOTES IN	D0710	COBOL
LOADER	NEW MCP OPTIONS IN	D0676	LOADER

KWIC SUBJECT INDEX

KWIC	SYSTEM NOTE	FUNCTION
LOADER ONLY LOADS MCPCODE FILE	D0614	LOADER
LOADING BX383 FIRMWARE	D0645	SCR
LOADS MCPCODE FILE LOADER ONLY	D0614	LOADER
LOCATOR TYPE INCREMENT	D0569	MAKEUSER
LOG CODES FOR LSP REMOVE	D0639	SCR
LOGGING SETSTATUS	D0593	MCP
LOGGING CHANGES	D0547	MCP
LOGGING CHANGES	D0639	SCR
LOGPOSITIONING OPTION	D0548	MCP
LOOPING ON USER OUTPUT SORT	D0623	SORT
LSP REMOVE LOG CODES FOR	D0639	SCR
MANAGEMENT ENHANCED DATA	D0572	MCP
MATRIX I-O	D0585	BASIC
MAXIMUM MCS"S LIMIT	D0561	NDL
MCP OPTIONS IN LOADER NEW	D0676	LOADER
MCPCODE FILE LOADER ONLY LOADS	D0614	LOADER
MCS ANALYSIS	D0627	DUMPANALY
MCS HANDLING	D0602	MCP-DATACM
MCS REQUESTS DIAGNOSTIC	D0604	DCPPROGEN
MCS"S LIMIT MAXIMUM	D0561	NDL
MEMDUMP IMPROVEMENTS	D0616	MCP
MEMORY INTEGRAL CALCULATION	D0621	MCP
MESSAGE B9352 ETX IN	D0626	SOURCENDL
MESSAGE CHANGES CONTROLLER	D0562	CONTROLLER
MESSAGES NEW SYSTEM	D0543	MCP
MISLEADING RESULTS FLAG	D0671	PACKDIR
MODEL 2741 TERMINAL TABS FOR	D0625	SOURCENDL
MODS ON LINE COUNT TOTAL	D0609	DUMPANALY
MOVE STATEMENTS	D0687	COBOL
MOVES CROSS-CLASS COMPARES AND	D0725	COBOL
MSG SPACE ALLOCATION CHANGE	D0622	NDL
MYJOB	D0670	MCP
MYSELF TASK	D0650	PLI

KWIC SUBJECT INDEX

KWIC	SYSTEM NOTE	FUNCTION
NAMES . CONDITION	D0660	COBOL
NAMES STRING VARIABLE	D0586	BASIC
NDL FILES BEFORE 2.6	D0716	MCP-DATACM
NDL SCANNER PROCEDURES	D0560	NDL
NEW ESPOL CONSTRUCT, CONVERT	D0591	ESPOL
NEW MCP OPTIONS IN LOADER	D0676	LOADER
NEW SYSTEM MESSAGES	D0543	MCP
NEW TASK ATTRIBUTES	D0613	WFL
NEW U FORMAT	D0681	ESPOLINTRN
NEW XREF FORMAT	D0654	PLI
NIFPREFIX SIZE	D0600	MCP-DATACM
NO-FILE TERMINATES NORMALLY	D0668	PACKDIR
NORMALLY NO-FILE TERMINATES	D0668	PACKDIR
NOW OPTIONAL CODEFILE	D0576	DUMPANALY
NOWWARN ADD DOLLAR OPTION	D0667	NDL
NUMBER PRINT TAPE SERIAL	D0610	DUMPANALY
ODD LENGTH RECORDS FILES WITH	D0683	BASIC
OFFSET IN SPOUT-DISPLAY CHAR	D0644	SCR
ON DISK FILES PATCH DECKS	D0634	PATCH
ON EMPTY LINE DCP RESULTS	D0601	MCP-DATACM
ON LINE COUNT TOTAL MODS	D0609	DUMPANALY
ON PACK BACKUP FILES	D0631	BACKUP
ON SEQ CHECK IGNORE INCL CARDS	D0715	BACKUP
ON SIMPLE STMTS WFL PRINTOUT	D0717	MCP
ON THREE CPU SYSTEMS CM	D0615	MCP
ON USER OUTPUT SORT LOOPING	D0623	SORT
ON 2.7 READ-ONLY DCPCODE FILES	D0726	NDL
ONLY LOADS MCPCODE FILE LOADER	D0614	LOADER
OPERATOR UNARY PLUS	D0578	WFL
OPERATORS RELATIONAL	D0588	BASIC
OPTION STMTNO	D0652	PLI
OPTION TIME DOLLAR	D0700	COBOL
OPTION CTMON DOLLAR	D0659	ALGOL

KWIC SUBJECT INDEX

KWIC SUBJECT INDEX

KWIC SUBJECT INDEX

KWIC SUBJECT INDEX

KWIC SUBJECT INDEX

KWIC	SYSTEM NOTE	FUNCTION
TAPES B5700 BACKUP	D0678	BACKUP
TASK MYSELF	D0650	PLI
TASK ATTRIBUTE ASSIGN AND REF	D0653	PLI
TASK ATTRIBUTES	D0581	MCP
TASK ATTRIBUTES NEW	D0613	WFL
TASK PRIORITY	D0571	MCP
TERMINAL TABS FOR MODEL 2741	D0625	SOURCENDL
TERMINATES NORMALLY NO-FILE	D0668	PACKDIR
TEST CLUSTER ALL ORDERS	D0665	DIAGNOSTMCS
THREE CPU SYSTEMS CM ON	D0615	MCP
TIME DOLLAR OPTION	D0700	COBOL
TOTAL MODS ON LINE COUNT	D0609	DUMPANALY
TRAIN PRINTERS	D0574	MCP
TRAIN PRINTERS RECOGNITION OF	D0566	LOADER
TRANSLATE ATTRIBUTES	D0658	ALGOL
TRANSLATION FULL SOFTWARE	D0594	MCP-I-O
TWO DIMENSIONAL ARRAYS	D0685	COBOL
TYPE INCREMENT LOCATOR	D0569	MAKEUSER
U FORMAT NEW	D0681	ESPOLINTRN
UNARY PLUS OPERATOR	D0578	WFL
UNIT-DEFAULT BUFFERS FLAG	D0636	SCR
UP-TAPE FILES	D0674	BACKUP
USASI OPTION - COMP ITEMS	D0721	COBOL
USE OF KINDLISTS SYSTEM	D0595	MCP-I-O
USER LABEL RECORDS	D0702	COBOL
USER OUTPUT SORT LOOPING ON	D0623	SORT
USER SEGMENTATION	D0682	ALGOL
USERDATA CHANGES	D0549	MCP
V FORMAT G AND	D0680	ESPOLINTRN
VARIABLE "AI" ADDED AS BYTE	D0606	DCPPROGEN
VARIABLE CRC[0],CRC[1] AS BYTE	D0575	NDL
VARIABLE NAMES STRING	D0586	BASIC
VARIABLE RUN PARAMETERS	D0648	SCR

KWIC SUBJECT INDEX

KWIC		SYSTEM NOTE	FUNCTION
VERSION	PAGE AND	D0624	SOURCENDL
VINTAGE	SOURCENDL	D0599	MCP-DATACM
VINTAGE	SOURCENDL	D0607	DIAGNOSTMCS
VOIDT	$GOTO WITH	D0613	PATCH
WAIT STMT	REWIND AND	D0647	SCR
WFL	EXTERNAL DECKS IN	D0629	MCP
WFL INTERFACE	RJE -	D0559	RJE
WFL PRINTOUT ON SIMPLE STMTS		D0717	MCP
WITH "LOW-VALUE"	EXAMINE	D0714	COBOL
WITH ODD LENGTH RECORDS	FILES	D0683	BASIC
WITH PUNCH OUTPUT	JOBS	D0628	MCP
WITH VOIDT	$GOTO	D0613	PATCH
WITHOUT "$"	CONTROL INPUT	D0713	MAKEUSER
WRITE BLOCK		D0693	COBOL
WRITE STMT FOR REMOTE FILES		D0701	COBOL
XREF FORMAT	NEW	D0654	PLI
2.6	NDL FILES BEFORE	D0716	MCP-DATACM
2.7 READ-ONLY	DCPCODE FILES ON	D0726	NDL
2741 TERMINAL	TABS FOR MODEL	D0625	SOURCENDL