B5700
SYSTEM NOTE 14

MARK 16 SYSTEM RELEASE

TABLE OF CONTENTS

} STAPLED SEPARATELY

B5500/B5700 SYSTEM SOFTWARE

MARK XVI.0.0 SYSTEM RELEASE

THE MARK XVI.0.0 SYSTEM RELEASE CONSISTS OF IMPROVEMENTS AND REFINEMENTS WHICH ENHANCE AND/OR CORRECT THE B5700 PROGRAMMING SYSTEMS.

THIS RELEASE CONSISTS OF THREE REELS OF MAGNETIC TAPE WHICH CONTAIN:

1. SYMBOLIC FILES OF ALL MARK XVI.0.0 SOFTWARE
2. OBJECT CODE VERSIONS OF ALL MARK XVI.0.0 SOFTWARE
3. PATCHES FROM MARK XV.3.0 TO MARK XVI.0.0
4. TEMPORARY PATCHES APPLICABLE TO THE MARK XVI.0.0 SYSTEM
5. PRINTER BACK-UP DISK FILES OF:
   A. THIS SYSTEM NOTE WHICH INCLUDES:
      1. APPENDIX A - NEW LIBRARY MAINTENANCE
      2. APPENDIX B - CHANGES TO "UPDATE/USERS"

NOTE: THE PATCHES WHICH WERE USED TO CREATE THE MARK XVI.0.0 SYMBOLICS ARE INCLUDED IN THIS RELEASE FOR DOCUMENTATION PURPOSES ONLY.

THERE ARE SEVERAL NEW FEATURES IN THIS RELEASE, THESE INCLUDE:

1. JULIAN DATE ADDED TO H/L MESSAGE.

2. DUMPCORE TAPE PARITY RECOVERY.

3. RC KEYBOARD REQUEST MESSAGE.

   A. WRITE PARITY REEL SWITCHING FACILITY.

   B. SYNTAX IS RC <MT> <UNIT NUMBER> EG. RCMTA.

4. AS AND QS REPLACE SM AND RS KEYBOARD REQUESTS.

   A. AS REPLACES SM (ACTIVITY SUMMARY).

   B. QS REPLACES RS (QUICK SS).

5. COMPILE TIME MONITOR OPTION MADE INDEPENDENT OF AUXMEM.

6. SEQUENCE ERROR COUNT TOTAL PRINTED AT END OF ALGOL AND/OR XALGOL COMPILATION LISTING IF CHECK OPTION IS SET.

7. SEQUENCE ERRORS AND SYNTAX ERRORS MADE MORE VISIBLE ON COMPILATION LISTINGS FOR ALGOL AND XALGOL.

8. B5700 NEW LIBRARY MAINTENANCE. SEE APPENDIX A.

9. CHANGES TO "UPDATE/USERS"

PAGE (3) RESERVED FOR THE TABLE OF CONTENTS

PAGE (4) RESERVED FOR THE TABLE OF CONTENTS

CONTENTS OF THE MARK XVI.0.0 RELEASE

CONTENTS OF THE MARK XVI.0.0 RELEASE

SYMBOL1/FILE000

~~THE "SYMBOL1/FILE000" TAPE IS A LIBRARY MAINTENANCE FORMAT "DUMP"~~
TAPE WHICH CONTAINS THE FOLLOWING SYMBOLIC FILES:

FILE NAME	DESCRIPTION
SYMBOL/MCP	ESPOL SYMBOLIC ** DATACOM MCP
SYMBOL/TSSMCP	ESPOL SYMBOLIC ** TIME-SHARING MCP
~~SYMBOL/INTRINS~~	~~ESPOL SYMBOLIC ** SYSTEM INTRINSICS MCP~~
SYMBOL/ALGOL	ALGOL SYMBOLIC ** ALGOL AND TSPOL COMPILERS
SYMBOL/COBOL	ALGOL SYMBOLIC ** COBOL COMPILER
~~SYMBOL/COBOL68~~	~~ALGOL SYMBOLIC ** COBOL68 COMPILER~~
SYMBOL/ESPOL	ALGOL SYMBOLIC ** ESPOL COMPILER
SYMBOL/XALGOL	ALGOL SYMBOLIC ** XALGOL COMPILER

SYMBOL2/FILE000

THE "SYMBOL2/FILE000" TAPE IS A LIBRARY MAINTENANCE FORMAT "DUMP" TAPE WHICH CONTAINS THE FOLLOWING SYMBOLIC FILES:

FILE NAME	DESCRIPTION
SYMBOL/FORTRAN	ALGOL SYMBOLIC ** FORTRAN COMPILER
SYMBOL/BASIC	ALGOL SYMBOLIC ** BASIC COMPILER
SYMBOL/AFILTER	ALGOL SYMBOLIC ** ALGOL FILTER PROGRAM
SYMBOL/AUXDATA	ALGOL SYMBOLIC ** AUXDATA/MAKER
SYMBOL/AUXTST	ESPOL SYMBOLIC ** AUXILIARY MEMORY TEST
SYMBOL/CHECKAL	ALGOL SYMBOLIC ** CHECKAL/TEST
SYMBOL/COOL	ESPOL SYMBOLIC ** COOL AND COLD START ROUTINES
SYMBOL/DC1000	ALGOL SYMBOLIC ** DC1000/CODEGEN
SYMBOL/DCFILL	ALGOL SYMBOLIC ** DCFILL/PRT
SYMBOL/DSKDSK	ESPOL SYMBOLIC ** DISK TO DISK LOADER
SYMBOL/DUMPANL	ALGOL SYMBOLIC ** DUMP/ANALYZE
SYMBOL/KERNEL	ESPOL SYMBOLIC ** HALT LOAD KERNEL ROUTINE
SYMBOL/LOGAN	ALGOL SYMBOLIC ** LOGAN/DISK
SYMBOL/LOGOUT	ALGOL SYMBOLIC ** LOGOUT/DISK
SYMBOL/LOGOUTR	ALGOL SYMBOLIC ** LOGOUTR/DISK
SYMBOL/MAKCAST	ALGOL SYMBOLIC ** MAKCAST/DISK
SYMBOL/MASTEST	COBOL SYMBOLIC ** MASTER/TEST
SYMBOL/MEMDUMP	ESPOL SYMBOLIC ** MEMORY DUMP ROUTINE
SYMBOL/MESSGEN	ALGOL SYMBOLIC ** SYSTEM/MESSGEN
SYMBOL/MLOGAN	ALGOL SYMBOLIC ** LOGANL/MAINT
SYMBOL/OLMAINT	TSPOL SYMBOLIC ** ONLINE/MAINT
SYMBOL/PMERGE	XALGOL SYMBOLIC ** PATCH/MERGE
SYMBOL/ROTO	ALGOL SYMBOLIC ** ROTO/ROOTER
SYMBOL/STATS1	ALGOL SYMBOLIC ** STATS1/ANALYZE
SYMBOL/STATS2	ALGOL SYMBOLIC ** STATS2/ANALYZE
SYMBOL/STATS3	ALGOL SYMBOLIC ** STATS3/ANALYZE
SYMBOL/STATS4	ALGOL SYMBOLIC ** STATS4/ANALYZE
SYMBOL/SYSDISK	ALGOL SYMBOLIC ** SYSDISK/MAKER
SYMBOL/TAPEDSK	ESPOL SYMBOLIC ** TAPE TO DISK LOADER
SYMBOL/TPECNF	OLMAINT SYMBOLIC ** OTPECNF/MAINT
SYMBOL/TSDUMP	ALGOL SYMBOLIC ** TSDUMP/ANALYZE
SYMBOL/TSFILL	ALGOL SYMBOLIC ** TSFILL/PRT
SYMBOL/UPDATE	ALGOL SYMBOLIC ** UPDATE/USERS
SYMBOL/CANDE	TSPOL SYMBOLIC ** CANDE/TSHARER
SYMBOL/APPEND	TSPOL SYMBOLIC ** APPEND/CANDE
SYMBOL/COPY	TSPOL SYMBOLIC ** COPY/CANDE
SYMBOL/DELETE	TSPOL SYMBOLIC ** DELETE/CANDE

```
SYMBOL/FIND       TSPOL SYMBOLIC ** FIND/DISK
SYMBOL/GUARD      TSPOL SYMBOLIC ** GUARD/DISK
SYMBOL/HARD       TSPOL SYMBOLIC ** HARD/CANDE
SYMBOL/HELP       TSPOL SYMBOLIC ** HELP/DISK
SYMBOL/LFILES     TSPOL SYMBOLIC ** LFILES/CANDE
SYMBOL/LIST       TSPOL SYMBOLIC ** LIST/CANDE
SYMBOL/LOAD       TSPOL SYMBOLIC ** LOAD/CANDE
SYMBOL/MERG       TSPOL SYMBOLIC ** MERGE/CANDE
SYMBOL/PAPER      TSPOL SYMBOLIC ** PAPER/CANDE
SYMBOL/PUNCH      TSPOL SYMBOLIC ** PUNCH/CANDE
SYMBOL/QUIKLST    TSPOL SYMBOLIC ** QUIKLST/CANDE
SYMBOL/REPLACE    TSPOL SYMBOLIC ** REPLACE/CANDE
SYMBOL/RESEQ      TSPOL SYMBOLIC ** RESEQ/CANDE
SYMBOL/RESEQB     TSPOL SYMBOLIC ** RESEQB/CANDE
SYMBOL/SCHEDUL    TSPOL SYMBOLIC ** SCHEDUL/CANDE
SYMBOL/USER       ALGOL SYMBOLIC ** USER/CANDE
```

SYSTEM/FILE000

THE FOLLOWING IS A LIST OF THE OBJECT CODE AND DATA FILES LOCATED ON THE TAPE LABELED "SYSTEM/FILE000":

MCP RELATED FILES

FILE NAME	DESCRIPTION
MCP/DISK	DCMCP WITH THE FOLLOWING OPTIONS SET TRUE: AUTODUMP BREAKOUT B6500LOAD CHECKLINK DATACOM DCLOG DCSPO DFX DISKLOG DUMP NEWLOGGING PACKETS SAVERESULTS SEPTICTANK WORKSETMONITOR WORKSET AND THE FOLLOWING OPTIONS SET FALSE: AUXMEM DEBUGGING DKBNODFX MONITOR SHAREDISK STATISTICS RJE
MCPA/DISK	DCMCP WITH THE SAME OPTIONS SET AS ABOVE: (TWO MCP-S ARE PROVIDED FOR LOADING PURPOSES.)
MCP/STUFF	DATA MCP "STUFF" FILE
MCP/PRT	DATACOM MCP "PRT" FILE

DC/AUXMCP	DATACOM MCP AUXILIARY MEMORY FILE
INT/DISK	DATACOM INTRINSICS OBJECT CODE FILE
INT/STUFF	DATACOM INTRINSICS "STUFF" FILE
DC/AUXINT	DATACOM INTRINSICS AUXILIARY MEMORY FILE
TSS/MCP	TIME-SHARING MCP WITH THE FOLLOWING OPTIONS SET TRUE: AUTODUMP B6500LOAD CHECKLINK DFX DUMP NEWLOGGING PACKETS SAVERESULTS SEPTICTANK AND THE FOLLOWING OPTIONS SET FALSE: AUXMEM DKBNUDFX MONITOR SHAREDISK STATISTICS TWXONLY
TSS/MCPA	TIME-SHARING MCP WITH THE SAME OPTIONS SET AS ABOVE: (TWO MCP-S ARE PROVIDED FOR LOADING PURPOSES.)
TSSMCP/STUFF	TIME-SHARING "STUFF" FILE
TSS/PRT	TIME-SHARING "PRT" FILE
TSS/AUXMCP	TIME-SHARING AUXILIARY MEMORY FILE
TSS/INT	TIME-SHARING INTRINSICS OBJECT CODE FILE
TSSINT/STUFF	TIME-SHARING INTRINSICS "STUFF" FILE
TSS/AUXINT	TIME-SHARING INTRINSICS AUXILIARY MEMORY FILE

COMPILER RELATED FILES

FILE NAME	DESCRIPTION
ALGOL/DISK	ALGOL COMPILER
BASIC/DISK	BASIC COMPILER
COBOL/DISK	COBOL COMPILER
COBOL68/DISK	COBOL68 COMPILER
ESPOL/DISK	ESPOL COMPILER

FORTRAN/DISK	FORTRAN COMPILER
MAKCAST/DISK	SYMBOLIC LIBRARY MAINTENANCE PROGRAM
TSPOL/DISK	TSPOL COMPILER
XALGOL/DISK	XALGOL COMPILER

CANDE RELATED FILES

FILE NAME	DESCRIPTION
CANDE/TSHARER	CANDE COMMAND AND EDIT PROGRAM
APPEND/CANDE	CANDE PROGRAM FOR APPEND VERB
COPY/CANDE	CANDE PROGRAM FOR COPY VERB
DELETE/CANDE	CANDE PROGRAM FOR DELETE VERB
FIND/DISK	CANDE PROGRAM FOR FIND VERB
GUARD/DISK	CANDE PROGRAM FOR GUARD VERB
HARD/CANDE	CANDE FILE MAINTENANCE PROGRAM
HELP/DISK	CANDE DISK FILE ERROR RECOVERY PROGRAM
LFILES/CANDE	CANDE PROGRAM FOR LFILES VERB
LIST/CANDE	CANDE PROGRAM FOR LIST VERB
LOAD/CANDE	CANDE PROGRAM FOR LOAD VERB
MERGE/CANDE	CANDE PROGRAM FOR MERGE VERB
MESAGE/CANDE	CANDE ERROR MESSAGE FILE
PAPER/CANDE	CANDE PROGRAM FOR PAPER VERB
PUNCH/CANDE	CANDE PROGRAM FOR PUNCH VERB
QUIKLST/CANDE	CANDE PROGRAM FOR QUIKLST VERB
REPLACE/CANDE	CANDE PROGRAM FOR REPLACE VERB
RESEQ/CANDE	CANDE PROGRAM FOR RESEQ VERB
RESEQB/CANDE	CANDE PROGRAM FOR RESEQB VERB
SCHEDUL/CANDE	CANDE PROGRAM FOR SCHEDUL VERB

SYSTEM UTILITY FILES

FILE NAME	DESCRIPTION
USER/CANDE	UPDATES "USERS/CANDE"
UPDATE/USERS	UPDATES "REMOTE/USERS"
PATCH/MERGE	PATCH MAINTENANCE PROGRAM
SYSDISK/MAKER	CREATES THE FILE "SYSTEM/DISK"
AUXDATA/MAKER	AUXILIARY MEMORY FILE MAINTENANCE PROGRAM
SYSTEM/MESSGEN	CREATES "MESSAGE/OTHEDAY"
DC1000/CODEGEN	GENERATES DC1000 R.J.E. CODE DECK

ANALYSIS RELATED FILES

FILE NAME	DESCRIPTION
ROTO/ROOTER	"SEPTIC" FILE ANALYZER
STATS1/ANALYZE	ANALYZER FOR TIME-SHARING STATISTICS FILE
STATS2/ANALYZE	ANALYZER FOR TIME-SHARING LOG WORKS ON LOG/DISK TYPE FILES
STATS3/ANALYZE	ANALYZER FOR DATACOM MCP STATISTICS FILE
STATS4/ANALYZE	ANALYZER FOR DATACOM MCP SYSTEM LOG
ONLINE/MAINT	ON-LINE MAINTENANCE PROGRAM
OTPECNF/MAINT	SET OF ON-LINE TAPE CONFIDENCE ROUTINES
LOGAN/DISK	TIME-SHARING LOG ANALYZER
LOGANL/MAINT	MAINTENANCE LOG ANALYZER
LOGOUT/DISK	DATACOM MCP SYSTEM LOG ANALYZER
LOGOUTR/DISK	DATACOM MCP REMOTE LOG ANALYZER
DCFILL/PRT	CREATES "MCP/PRT"
DUMP/ANALYZE	DATACOM MCP MEMORY DUMP ANALYZER
TSFILL/PRT	CREATES "TSS/PRT"
TSDUMP/ANALYZE	TIME-SHARING MCP MEMORY DUMP ANALYZER

PUNCH BACK-UP FILES OF "CARD LOAD SELECT" PROGRAMS

FILE NAME	DESCRIPTION
PUD/COLD	COLD START PROGRAM
PUD/COOL	COOL START PROGRAM
PUD/KERNEL	HALT LOAD KERNEL ROUTINE
PUD/MEMDUMP	MEMORY DUMP ROUTINE
PUD/DSKDSK	DISK TO DISK PROGRAM
PUD/TAPEDSK	TAPE TO DISK PROGRAM
PUD/AUXTST	AUXILIARY MEMORY TEST

GENERAL UTILITY FILES

FILE NAME	DESCRIPTION
CHECKAL/TEST	ALGOL MASTER TEST PROGRAM
MASTER/TEST	COBOL MASTER TEST PROGRAM
TAPE/COMPARE	TAPE COMPARISON PROGRAM
TAPCOPY/DISK	TAPE COPY AND COMPARISON PROGRAM
DSKDUMP/UTILITY	LISTS DISK AREAS BY ADDRESS
HDRLST/UTILITY	LISTS DISK DIRECTORY HEADERS BY NAME
LIBLST/UTILITY	LIST SYMBOLIC FILES ON "LIBRARY DUMP" TAPES
XREF/JONES	CROSS REFERENCE AND DOCUMENT EDITING PROGRAM

PATCHES TO MARK XV.3.0

FILE NAME
MARKXVI/MCP
MARKXVI/TSSMCP
MARKXVI/INTRINS
MARKXVI/ALGOL
MARKXVI/BASIC
MARKXVI/COBOL
MARKXVI/COBOL68
MARKXVI/ESPOL
MARKXVI/FORTRAN
MARKXVI/XALGOL
MARKXVI/CANDE
MARKXVI/APPEND
MARKXVI/COPY
MARKXVI/DELETE
MARKXVI/FIND
MARKXVI/GARD
MARKXVI/HARD
MARKXVI/HELP
MARKXVI/LFILES
MARKXVI/LIST
MARKXVI/LOAD
MARKXVI/MERG
MARKXVI/PAPER
MARKXVI/PUNCH

MARKXVI/QUIKLST
MARKXVI/REPLACE
MARKXVI/RESEQ
MARKXVI/RESEQB
MARKXVI/SCHEDUL
MARKXVI/UPDATE
MARKXVI/USER
MARKXVI/PMERGE
MARKXVI/SYSDISK
MARKXVI/AUXDATA
MARKXVI/MESSGEN
MARKXVI/ROTO
MARKXVI/STATS1
MARKXVI/STATS2
MARKXVI/STATS3
MARKXVI/STATS4
MARKXVI/OLMAINT
MARKXVI/TPECNF
MARKXVI/LOGAN
MARKXVI/MLOGAN
MARKXVI/LOGOUT
MARKXVI/LOGOUTR
MARKXVI/DCFILL
MARKXVI/DUMPANL
MARKXVI/TSFILL
MARKXVI/TSDUMP
MARKXVI/COOL
MARKXVI/KERNEL
MARKXVI/MEMDUMP
MARKXVI/DSKDSK
MARKXVI/TAPEDSK
MARKXVI/AUXTST
MARKXVI/CHEKAL
MARKXVI/MASTEST
MARKXVI/MAKCAST
MARKXVI/AFILTER

TEMPORARY PATCHES TO MARK XVI.0.0

FILE NAME
~~PATCH/MCP~~
PATCH/TSSMCP
PATCH/INTRINS
~~PATCH/ALGOL~~
PATCH/BASIC
PATCH/COBOL
~~PATCH/COBOL68~~
PATGH/FORTRAN
PATCH/ESPOL
~~PATCH/XALGOL~~
PATCH/CANDE
PATCH/COOL
~~PATCH/TSFILL~~
PATCH/TSDUMP
PATCH/DCFILL
~~PATCH/DUMPANL~~
PATCH/ROTO
PATCH/MAKCAST

PRINTER BACK-UP FILES

FILE NAME	DESCRIPTION
PBD/SYSNOTE	SYSTEM NOTE

CUMULATIVE DOCUMENTATION REFERENCE

APPENDICES TO SYSTEM NOTES

1 (XI.0)	A	~~FILE ATTRIBUTES~~
	B	FILE PARAMETER BLOCK LAYOUT
	C	DISK ORGANIZATION
	D	~~I/O ERROR MESSAGES~~
	E	COLD AND COOL START DECK CONSTRUCTION
	F	SYSTEM MEASUREMENT FACILITIES ~~(STATISTICS COMPILE TIME OPTION)~~
	G	TAPCOPY/DISK OPERATING INSTRUCTIONS
4 (XII,0)	A	~~RESOURCE ALLOCATION AND AUXMEM~~
	B	CHANGES AND ADDITIONS TO CANDE VERBS
	C	EXTENDED DATACOM FOR TIME SHARING
	D	~~USE OF THE B9352 WITH TIME SHARING~~
	E	USE OF THE TC500 WITH TIME SHARING
5 (XII,24)	A	~~NEW CANDE FEATURES~~
	B	ON-LINE MAINTENANCE MANUAL
6 (XII,98)	A	~~PUNCH BACK-UP FACILITY~~
	B	PATCH/MERGE USERS GUIDE
7 (XIII.0)	A	~~REMOTE JOB ENTRY~~
	B	ON-LINE MAINTENANCE TAPE ROUTINES
9 (XIII.69)	A	~~ALGOL/XALGOL DOLLAR CARD SYNTAX~~
	B	COBOL FILE ATTRIBUTES
	C	COBOL68 INTER-PROGRAM COMMUNICATION
	D	~~FORTRAN DOLLAR CARD SYNTAX~~
	E	MAINTENANCE LOG MANUAL
	F	ONLINE/MAINT - SIMPL MANUAL
10 (XIV.0)	A	REAL-TIME TAPE TEST FACILITY
	B	DCMCP PACKETS OPTION
	C	~~COBOL68 FILE ATTRIBUTES~~
	D	AUXMEM ON TIME SHARING
	E	MULTI-REEL LIBRARY TAPES
	F	~~TC500/CANDE INTERFACE PROGRAM~~
	G	B9353 WITH TIME SHARING
	H	ONLINE/MAINT CHANGES
	I	~~SUPPLEMENT TO SIMPL~~
	J	COBOL68 EVENTS AND INTERRUPTS
	K	FORTRAN FORMAT IMPROVEMENTS
11 (XV.1.0)	A	ESPOL COMPILER CONTROL CARDS

	B	AUXILIARY MEMORY DESCRIPTION
	C	~~"PACKETS" FOR TSSMCP~~
	D	AUXILIARY MEMORY TEST FACILITY
	E	DISKSQUASH FACILITY
	F	~~CHANGES TO "UPDATE/USERS"~~
	G	THE DATA COMM TRACKING FACILITY
	H	DIRECTORY INTERLOCKING DESCRIPTION
	I	~~REVISED HANDLING OF DISK ERRORS~~
	J	PBD/PUD RECOVERY AFTER A HALT/LOAD
	K	MEMORY DUMP CHANGES
	L	~~DISK SPEED AND/OR LU ATTRIBUTES~~
	M	"XREF" AND "BEND" OPTIONS IN ALGOL
	N	ONLINE DISK CONFIDENCE TEST
	O	~~CANDE "HELP" ROUTINE~~
12 (XV,2,0)	A	THE SENSITIVE ATTRIBUTE
	B	~~NEW COOL/COLD START FEATURES~~
	C	NO MEM AIDS
	D	AUXMEM RECOVERY
	E	~~RECORD LEVEL LOCKOUT~~
	F	PRINTER BACK-UP MODIFICATIONS
~~13 (XV,3,0)~~	~~A~~	~~WORKING SET~~
14 (XVI,0)	A	NEW LIBRARY MAINTENANCE
	B	~~CHANGES TO "UPDATE/USERS"~~

MANUALS IN PBD FORMAT ON SOFTWARE RELEASES

RELEASE	MANUAL
MARK XIII SYSTEM RELEASE (MARK XIII.0 MCP)	TIME SHARING SYSTEM REFERENCE MANUAL
MARK XIII PATCH REL. #1 (MARK XIII.69 MCP)	COBOL68 (CODASYL) MANUAL B5700 MCP REFERENCE MANUAL
MARK XV.2.0 SYSTEM RELEASE (MARK XV.2.0 MCP)	SYSTEM OPERATIONS GUIDE

TO OBTAIN ADDITIONAL DOCUMENTATION, PLEASE REFER TO THE "ELECTRONIC DATA PROCESSING (GROUP III) PRINTED MATERIALS CATALOG AND PRICE LIST" PUBLICATION NUMBER: 1047800

CHANGES TO THE MARK XV.3.0 SYSTEM

MCP CHANGES 1 TROUGH 57.

CHANGE NO. 1 (143 CARDS).

TERMINATION AND STOPPING

THIS CHANGE CORRECTS POTENTIAL DEADLOCK SITUATIONS ARISING FROM USE OF THE GLOBALS "TERMIX" (<MIX>DS) AND "STOPJOB" (<MIX>ST). THESE GLOBALS HAVE BEEN ELIMINATED IN FAVOR OF A SYSTEM SOMEWHAT LIKE THAT CURRENTLY USED IN THE TSSMCP. AS IN THE TSSMCP, BITS ARE KEPT IN PRTROW[MIX].[3:4] WHICH INDICATE ACTION TO BE TAKEN BY VARIOUS TERMINATION AND STOPJOB ROUTINES.

[6:1] = 1 INDICATES THE JOB IS MARKED FOR EVENTUAL TERMINATION
[3:4] = 2 INDICATES THE JOB IS MARKED FOR EVENTUAL STOPPING
[3:4] = 3 INDICATES THE JOB HAS PASSED THE TERMINAL MESSAGE STAGE IN THE TERMINATION PROCESS
[3:1] = 1 IS USED BY RESTART ROUTINES TO DETERMINE IF ITS SPECIAL TERMINATION ROUTINE "RSDSED" SHOULD BE INITIATED

THERE ARE SOME DIFFERENCES SO FAR AS INTERPRETATION OF THESE BITS IS CONCERNED BETWEEN TSSMCP AND THIS PATCH, WHICH WERE DUE TO THE FOLLOWING:

1. PRTROW[MIX] HAS NOT HAD ITS ROW FILLED WHEN A RESTART JOB IS BEGINNING; THEREFORE, IN TERMINATE WE CHECK THE JARROW ENTRY AS BEING NON-ZERO TO DETERMINE VALIDITY IN SETTING THE DS-ED BITS AT THAT POINT.
2. NO JOBS ARE BEING SWAPPED IN THE BATCH ENVIRONMENT, SO USE OF TERMINALCLOCK COULD BE RETAINED. ITS VALUE IS CHECKED IN NSECOND TO DETERMINE IF SOMEHOW THE JOB IS NOT BEING DS-ED. IF THAT IS SO, TERMINATION IS FORCED AND TERMINALCLOCK IS RESET. HOWEVER, THE PROCESS OF CLEANING UP BEFORE ENDING A JOB CAN BE QUITE LENGTHY, SO THE PROTECTIVE VALUE OF 3 IN PRTROW[MIX].[3:4] IS GIVEN AT THE START OF TERMINALMESSAGE SO THAT NSECOND WOULD NOT REINITIATE THE TERMINATION PROCESS ON THESE JOBS.

CHANGE NO. 2 (1 CARD).

NUMBER OF LINES IN QT MSG

THIS CHANGE CAUSES THE CORRECT NUMBER OF LINES QT-ED TO BE PRINTED IF MORE THAN 32768 LINES ARE SKIPPED.

CHANGE NO. 3 (1 CARD).

B6500 LOAD OF A B5500 TAPE

THIS CHANGE HANDLES A POSSIBLE INVALID LINK SITUATION ARISING WHEN TRYING TO PERFORM A "CC LOAD B6500 <ETC.>" ON A B5500 LIBRARY TAPE.

CHANGE NO. 4 (1 CARD).

REMOVE NAME, OPTN 2

THIS CHANGE REMOVES THE DUMMY NAME ASSOCIATED WITH OPTION WORD BIT # 2. THIS WORD IS USED IN CONNECTION WITH MOD 3 I/O'S AND MUST HAVE DEFINITION, BUT NOW IT WILL NOT APPEAR IF OPTIONS ARE LISTED.

CHANGE NO. 5 (10 CARDS).

USE OF NT1 IN CONTROLCARD

THIS CHANGE CORRECTS THE IMPROPER USE OF GLOBAL VARIABLES SUCH AS NT1 THROUGHOUT THE CONTROLCARD RELATED PROCEDURES. THE PROBLEMS AROSE AS A RESULT OF SPLITTING CONTROLCARD INTO MANY ROUTINES AND HAVING THE ROUTINES SHARE LOCAL VARIABLES.

CHANGE NO. 6 (45 CARDS).

PACKETS - ZIPARRAY

THIS CHANGE IMPROVES THE READABILITY OF CONTROL CARDS ZIPPED WITH ARRAY. ONE STATEMENT PER LINE IS LISTED ON THE PACKET PAGE. ALSO, THIS CHANGE ADDS THE PRORATED TIME TO THE JOB STATISTICS APPEARING ON THE PACKET PAGE FOR BATCH.

CHANGE NO. 7 (3 CARDS).

NEWLOGGING - 2 PROCESSORS

THIS CHANGE CORRECTS AN ERROR IN THE NEWLOGGING OPTION IN WHICH A JOB COULD BE INITIATED WITHOUT RESTARTING ITS PROCESSOR LOGGING CLOCK ON A 2 PROCESSOR SYSTEM.

CHANGE NO. 8 (1 CARD).

STARTIMING - SHAREDISK

THIS CHANGE CORRECTS AN ERROR WITH THE SHAREDISK CODE IN STARTIMING. THIS ERROR WOULD RESULT IN A SHIFTING OF THE PROGRAMS FPB BY ONE WORD.

CHANGE NO. 9 (98 CARDS).

NEWLOGGING -WM RESPONSE

THIS CHANGE ADDS THE "NEWLOGGING", "WORKSET" AND "WORKSETMONITOR" OPTIONS TO THE WM RESPONSE WHICH WERE LEFT OUT WHEN NEWLOGGING AND WORKSET WERE RELEASED.

CHANGE NO. 10 (198 CARDS).

NO MEM RECOVERY

WITH THIS CHANGE, FOR A PERIOD OF APPROXIMATELY TWO MINUTES AFTER THE OCCURRENCE OF A NO MEM CONDITION, MCP WORKING STORAGE IS OBTAINED AS NEAR TO THE FRONT OF MEMORY AS POSSIBLE. THIS REDUCES CHECKERBOARDING AND THEREFORE ENHANCES THE POSSIBILITY OF SYSTEM RECOVERY. THIS PATCH IS DESCRIBED IN DETAIL IN APPENDIX B OF THE SYSTEM NOTES FOR THE MARK XV.2 RELEASE.

CHANGE NO. 11 (7 CARDS).

FILE-IN-USE ERRORS

THIS CHANGE CORRECTS SEVERAL ERRORS IN PROCEDURE DIRECTORYSEARCH ASSOCIATED WITH FILE-IN-USE CONDITIONS. THE MOST COMMON SYMPTOM WAS THAT A PROGRAM WHICH ATTEMPTED TO ACCESS A FILE WHILE ITS NAME WAS BEING CHANGED WAS NOT AWAKENED WHEN THE NAME CHANGE WAS COMPLETED.

CHANGE NO. 12 (5 CARDS).

HDR SPACE LIBRARYZERO

THIS CHANGE CORRECTS AN ERROR WHICH OCCASIONALLY LEFT 30 WORDS OF MEMORY IN-USE AFTER LIBMAIN/DISK BLANKED A SENSITIVE FILE.

CHANGE NO. 13 (3 CARDS).

RRRMECH IN TAPE RETRY ON LABEL

THIS CHANGE CORRECTS AN ERROR IN THE INTERLOCKING BETWEEN THE MCP PROCEDURE STATUS AND OTHER PROCEDURES AND PROGRAMS WHICH ACCESS TAPE UNITS. THE ERROR OCCURRED WHEN A PARITY RETRY WAS DONE ON AN I/O TO THE TAPE LABEL. USUALLY, A SYSTEM HANG OCCURRED. THE ERROR WAS INTRODUCED IN PATCH XV.2.20 .

CHANGE NO. 14 (8 CARDS).

NEWLOGGING CORRECTIONS

THIS CHANGE CORRECTS AN ERROR IN WORKSET. IN THE PROCEDURE WORKSET AN INCORRECT TEST WAS MADE TO DETERMINE IF THE PROCESSOR TIME SHOULD BE INCREMENTED BY THE INTERNAL CLOCK. THIS COULD RESULT IN AN INCORRECT PROCESSOR TIME BEING USED IN THE PROCEDURE TO DETERMINE TOTAL PROCESSOR TIME BEING USED.

CHANGE NO. 15 (10 CARDS).

IDLETIME CORRECTION

THIS CHANGE CORRECTS AN ERROR IN WHICH THE IDLE TIME CHARGED TO A JOB WOULD BE INCORRECT DUE TO AN INCORRECT CALL ON THE PROCEDURE IDLETIME IN THE SELECTRUN ROUTINE.

CHANGE NO. 16 (5 CARDS).

CCSET HEADER SPACE RELEASE

THIS CHANGE CAUSES THE SYSTEM TO RELEASE THE HEADER FOR OTHER USE. PREVIOUSLY, WHEN A CONTROL CARD SET OR RESET A FILE "ACCESSED", "FIXED" OR "SENSITIVE" THE SYSTEM LEFT THE HEADER IN CORE.

CHANGE NO. 17 (5 CARDS).

MTA ACCESS AFTER DS DURING FM

THIS CHANGE CORRECTS AN ERROR WHICH CAUSED THE MCP TO ATTEMPT TO ACCESS MTA AFTER A JOB WHICH WAS WAITING FOR AN FM WAS DS-ED.

CHANGE NO. 18 (32 CARDS).

REASON ADDED TO SCHEDULED MESSAGE

THIS CHANGE CAUSES A MESSAGE TO BE SPOUTED IN ALL CASES WHEN A JOB IS NOT INITIALIZED AFTER A REQUEST TO XS OR ES IT. THE FORMAT OF THIS MESSAGE IS:

<PRIORITY>:<JOB SPECIFIER> NOT XS-ED,<REASON>

THIS REPLACES THE MESSAGE "NOT XS-ED (<REASON>)" WHICH WAS USED PREVIOUSLY. IN ADDITION, THE REASON FOR WHICH A JOB IS INITIALLY SCHEDULED HAS

<PRIORITY>:<JOB SPECIFIER> SCHEDULED <TIME><REMOTE INFORMATION>, <REASON>

WHERE <REMOTE INFORMATION> IS EITHER EMPTY OR "FROM <TERMINAL UNIT>/<BUFFER>" AND <REASON> IS ONE OF THE FOLLOWING:

NO MEM	WHICH MAY MEAN THAT A NO MEM CONDITION CURRENTLY EXISTS OR THAT THE CORE ESTIMATE ADDED TO THE CORE IN USE EXCEEDS THE SPACE AVAILABLE AS DETERMINED BY THE MULTI-PROCESSING FACTOR.
NO OLAY DISK	
TOO MANY JOBS	WHICH MEANS THAT MIXMAX JOBS ARE ALREADY EXECUTING
RESTART IN PROGRESS	WHICH IS BECAUSE ONLY ONE RESTART IS ALLOWED AT A TIME.

CHANGE NO. 19 (2 CARDS).

UNIT FIELD IN ACTUALIOERROR

THIS CHANGE RESTORES THE UNIT DESIGNATE FIELD IN THE RESULT DESCRIPTOR PASSED TO MCP PROCEDURE ACTUALIOERROR. THE PRIMARY PURPOSE OF THIS IS FOR COMPATIBILITY WITH THE ISSMCP.

CHANGE NO. 20 (2 CARDS).

SYNTAX ERR. PACKETS VS AUXMEM

THIS CHANGE CORRECTS A SYNTAX ERROR THAT OCCURRED WHEN THE MCP WAS COMPILED WITH THE PACKETS OPTION SET AND THE AUXMEM OPTION RESET.

CHANGE NO. 21 (5 CARDS).

PBD FILES, NUM > 8000, COPIES

THIS CHANGE CORRECTS 2 ERRORS IN THE HANDLING OF BACK UP DISK FILES:

1. IF THE NUMBER OF A PBD FILE WAS GREATER THAN OR EQUAL TO 8000 , IT WAS LEFT IN USE.
2. IF MORE THAN 2 COPIES WERE SPECIFIED FOR A FILE IN A PACKET ANY FILES FOLLOWING THAT ONE WERE NOT REMOVED AFTER THE

PACKET WAS PRINTED.

CHANGE NO. 22 (1 CARD).

INCORRECT PUNT

THIS CHANGE CORRECTS AN INCORRECT CALL ON THE PUNT PROCEDURE WHICH COULD HAVE RESULTED IN GIBBERISH BEING PRINTED AS THE <REASON> IN A "=SYS HANG" MESSAGE.

CHANGE NO. 23 (12 CARDS).

LOOP AFTER BUSY IN SMW

THIS CHANGE CORRECTS A SITUATION WHICH CAUSED THE MCP TO PRINT "BREAK" AT THE END OF A BUFFER OF INPUT AT A REMOTE SPO IF OUTPUT WERE QUEUED FOR THAT STATION WHILE THE INPUT WAS BEING ENTERED.

CHANGE NO. 24 (3 CARDS).

UVSPACE RETURNED IN SELECTRUN

THIS CHANGE CAUSES THE SIXTEEN WORDS OBTAINED FOR THE UV ARRAY FOR A JOB TO BE RETURNED IN THE CASE THAT THE JOB IS NOT INITIATED DUE TO LACK OF APPROPRIATE MEMORY SPACE FOR ITS STACK AND PRT. THIS CHANGE AFFECTS ONLY THOSE MCP=S WHICH INCLUDE THE STATISTICS COMPILE=TIME OPTION.

CHANGE NO. 25 (1 CARD).

CONTROL CRD ERR AT REMOTE STA

THIS CHANGE CORRECTS A CONDITION WHICH CAUSED THE CONTROL CARD ERROR MESSAGE FOR AN ERRONEOUS CONTROL CARD ENTERED AT A REMOTE STATION TO BE PRINTED AT THE SYSTEM SPO RATHER THAN AT THE REMOTE STATION.

CHANGE NO. 26 (4 CARDS).

RRRMECH ERR PB KBD REQUEST

THIS CHANGE ELIMINATES AN ERROR WHICH COULD OCCUR WHEN PRINTING OF A BACK UP TAPE WHICH WAS POSITIONED AT THE START OF THE REEL WAS INITIATED VIA A PB KEYBOARD REQUEST. THE ERROR CAUSED INCORRECT

QUEUEING IN THE I/O QUEUES AND THEREBY A SYSTEM HANG.

CHANGE NO. 27 (3 CARDS).

WORDS RQD ADDED TO SCHED MSG

THIS CHANGE ADDS THE NUMBER OF WORDS REQUIRED TO THE SCHEDULE MESSAGE IF A JOB IS SCHEDULED DUE TO A NO MEM CONDITION. THE FORMAT FOR THAT CASE IS NOW:
<PRIORITY>:<JOB SPECIFIER>SCHEDULED<TIME><REMOTE INFORMATION>,
NEEDS<NUMBER OF WORDS>

CHANGE NO. 28 (18 CARDS).

MEND, MSTART DELETED

THIS CHANGE ELIMINATES THE MCP GLOBAL VARIABLES MEND AND MSTART AND MAKES THE GLOBAL MEMASK DEPENDENT ON THE DUMP AND BREAKOUT OPTIONS. THIS IS DONE TO REDUCE THE SIZE OF THE MCP'S PRT.

CHANGE NO. 29 (22 CARDS).

FINDINPUT WAITS FOR CONTROLCRD

THIS CHANGE CORRECTS A CONDITION WHICH CAUSED A NO FILE FOR A PROGRAM WHEN, IN FACT, THE MCP WAS MERELY SLOW PROCESSING THE LABEL OF A CARD DECK. NOW, THE MCP WAITS UNTIL CONTROL CARDS ARE NOT BEING PROCESSED BEFORE TESTING THE LABEL OF A CARD READER.

CHANGE NO. 30 (16 CARDS).

MONITOR INDEPENDENT OF AUXMEM

THIS CHANGE CORRECTS AN ERROR WHICH PREVENTED INFORMATION FROM BEING COLLECTED WHEN THE MONITOR OPTION WAS SET. IN ADDITION, THE MONITOR OPTION IS NOW INDEPENDENT OF THE AUXMEM OPTION. THAT IS, AN MCP CAN BE COMPILED WITH MONITOR SET AND AUXMEM RESET.

CHANGE NO. 31 (32 CARDS).

PBD, TWO FILES TO ONE PRINTER

THIS CHANGE CORRECTS AN ERROR WHICH COULD CAUSE TWO FILES TO BE

PRINTED ON ONE PRINTER AT THE SAME TIME. THIS COULD OCCUR ONLY IF AUTOPRNT WAS SET.

CHANGE NO. 32 (24 CARDS).

DELAY COM MOVED TO CUMNICATED

THIS CHANGE MOVES THE CODE FOR COMMUNICATE 30 FROM PROCEDURE SHORTCOMMUNICATE TO PROCEDURE COMMUNICATED, THUS REDUCING THE SIZE OF THE FREQUENTLY USED SHORTCOMMUNICATE PROCEDURE.

CHANGE NO. 33 (3 CARDS).

H/L INTRINSIC INITIALIZATION

THIS CHANGE CORRECTS A CONDITION WHICH CAUSED THE MCP TO USE FILE 0000000/0000000, IF PRESENT, AS THE INTRINSICS WHEN A HALT/LOAD WAS DONE AT A TIME WHEN THE INTRINSIC FILE WAS NOT SPECIFIED, SUCH AS AFTER A COOL/START.

CHANGE NO. 34 (20 CARDS).

INVLD AUXMEM IO MESSAGE

THIS CHANGE CORRECTS THE MESSAGE WHICH IS SPOUTED IF AN AUXMEM I/O IS ATTEMPTED ON A SYSTEM WHICH DOES NOT INCLUDE THE AUXMEM OPTION. IF THIS OCCURS, THE <REASON> IN THE "SYS HANG" MESSAGE WILL BE INVLD AUXMEM IO

CHANGE NO. 35 (11 CARDS).

TOTAL CORE USED IF EST LARGER

THIS CHANGE CAUSES THE CORE ESTIMATE FOR A JOB WHOSE ESTIMATE IS GREATER THAN THE TOTAL CORE AS LISTED IN THE RESPONSE TO A TF REQUEST TO BE TAKEN AS THE TOTAL CORE. THIS WILL NORMALLY ENSURE THAT LDCNTRL/DISK AND PRNPBT/DISK WILL EXECUTE SINCE THEY ARE TESTED AGAINST A VALUE 10% GREATER THAN THE LISTED TOTAL CORE.

CHANGE NO. 36 (3 CARDS).

HANDLING OF PB MSG

THIS CHANGE CORRECTS TWO ERRORS INTRODUCED IN THE MARK XV.3 RELEASE. ONE OF THESE CAUSED PRNPBT/DISK TO ATTEMPT TO OUTPUT BACK-UP TAPES ON MTA TO THE PUNCH RATHER THAN THE PRINTER (IF A PB WERE USED TO START PRNPBT/DISK). THE OTHER CAUSED PRNPBT/DISK TO GO TO EOJ WITHOUT PUNCHING ANY CARDS IF A PB WERE USED TO PUNCH A BACKUP DISK FILE.

CHANGE NO. 37 (3 CARDS).

EXPECTED IO ERRS

THIS CHANGE ELIMINATES A POSSIBLE ERROR IN THE HANDLING OF "EXPECTED" I/O ERRORS. IT ALSO REDUCES THE SIZE OF THE IOFINISH PROCEDURE.

CHANGE NO. 38 (3 CARDS).

JOBMESS LINE FOR MCP/DISK

THIS CHANGE CORRECTLY POSITIONS THE INFORMATION FOR THE MCP WHICH IS OBTAINED IN RESPONSE TO SUCH REQUESTS AS CU AND AU.

CHANGE NO. 39 (2 CARDS).

NULL WY

THIS CHANGE ELIMINATES AN EXTRANEOUS CHARACTER THAT APPEARED IN FRONT OF THE WORD "NULL" WHEN PRINTED IN RESPONSE TO A WY KEYBOARD REQUEST.

CHANGE NO. 40 (2 CARDS).

PUNT FOR ESPDISK ERROR

THIS CHANGE CAUSES THE "ESPDISK ERROR" MESSAGE TO BE OUTPUT BY THE MCP PROCEDURE PUNT, THUS ALLOWING THE AUTODUMP CODE, IF PRESENT, TO BE INVOKED. PRIOR TO THIS PATCH, THE SYSTEM CONTINUED TO RUN AFTER AN ESPDISK ERROR, ALTHOUGH IT WAS INEVITABLE THAT A H/L BE DONE SOON.

CHANGE NO. 41 (1 CARD).

SYSTEM NUMBER IN SEARCH STMT

THIS CHANGE ADDS THE SYSTEM NUMBER TO THE 9:2 FIELD OF THE SIXTH WORD OF THE ARRAY USED IN A SEARCH STATEMENT. THIS CHANGE AFFECTS SHAREDISK SYSTEMS ONLY.

CHANGE NO. 42 (14 CARDS).

SHAREDISK,SQSTOP

THIS CHANGE CORRECTS THE PROCESSING OF AN SQSTOP WHEN ENTERED ON A SHAREDISK SYSTEM WHEN THE INITIATING SYSTEM IS WAITING FOR THE OTHER SYSTEMS TO BE SQ-ED.

CHANGE NO. 43 (2 CARDS).

DS JOB WITH BLOCK OVER 1890 WD

THIS CHANGE CAUSES A PROGRAM WHICH ATTEMPTS TO OPEN A DISK FILE FOR WHICH THE BLOCK SIZE IS GREATER THAN 1890 WORDS (63 SEGMENTS) TO BE DS-ED WITH THE MESSAGE:

INVALID BLOCK <MULTI-FILE ID>/<FILE ID> <RECORD SIZE><BLOCK SIZE> <SEGMENTS PER BLOCK>

CHANGE NO. 44 (13 CARDS).

RN# ACCEPTS LESS THAN 4 DIGITS

THIS CHANGE ALTERS THE PROCESSING OF THE RN MESSAGE TO ACCEPT FEWER THAN 4 DIGITS AFTER A #, THAT IS, RN#14 IS NOW ALLOWED IN ADDITION TO RN#0014.

CHANGE NO. 45 (4 CARDS).

JULIAN DATE IN WD AND H/L

THIS CHANGE ADDS THE JULIAN DATE TO THE RESPONSE TO A WD KEYBOARD REQUEST AND ALSO CAUSES IT TO BE PRINTED DURING A H/L. THE FORMAT IS:

DATE IS <MONTH>/<DAY>/<YEAR>=(<JULIAN DATE>)

CHANGE NO. 46 (11 CARDS).

REMOTE KEYBOARD REQUESTS

THIS CHANGE DOES THE FOLLOWING THINGS TO CORRECT ERRORS IN THE PROCESSING OF KEYBOARD REQUESTS RELATED TO REMOTE TERMINALS.

1. THE OPERATION OF THE TC MESSAGE IS RESTORED.
2. THE OPERATION OF THE <MIX>HM MESSAGE IS RESTORED.
3. EOJ MESSAGES ARE SUPPRESSED IF AN HM HAS BEEN DONE.
4. THE OPERATION OF THE SM MESSAGE IS RESTORED.
5. THE QV MESSAGE RETURNS THE CURRENT QV VALUE IF A NEW VALUE IS NOT INCLUDED IN THE REQUEST.

CHANGE NO. 47 (3 CARDS).

HEADER AND LEVEL CARDS

THIS CHANGE UPDATES THE HEADER AND LEVEL CARDS.

CHANGE NO. 48 (57 CARDS).

SYSTEM JOB CONTROL

THIS CHANGE MODIFIES THE WAY IN WHICH THE MCP CHECKS FOR THE SYSTEM JOBS LIBMAIN/DISK,LDCNTRL/DISK AND PRNPBT/DISK. RATHER THAN USE A SINGLE BIT IN JAR[9] AND TESTING ON THE NAME, A THREE BIT CODE FOR EACH IS USED. THIS IS ALSO IN JAR[9]. ALL THE CODES ARE ODD VALUES TO FACILITATE BOOLEAN TESTING; ZERO IS USED FOR ALL OTHER JOBS. THE VARIABLES JAR[9], SHEET[2] AND MCPJOB ARE AFFECTED.

CHANGE NO. 49 (13 CARDS).

LOCALIZE CONTROLCARD VARIABLES

THIS CHANGE MAKES THE F-RELATIVE VARIABLES WHICH ARE USED IN CONTROLCARD AND RELATED PROCEDURES (LIBCC,CCFINISH CCSET, CCCOMPIL5, INITIALIZEIT, CCUNIT, CCSECMAINT, CCLABEL, AND CCFIND) LOCAL TO E1C8 ONE INSTEAD OF GLOBAL TO THE WHOLE GROUP (AND THE REMAINDER OF THE MCP).

CHANGE NO. 50 (63 CARDS).

OVERLAY LIBMSG TABLE

THIS CHANGE REDUCES THE NUMBER OF DISK I/O-S WHICH NEED TO BE DONE TO USE THE TABLE OF LIBRARY MESSAGES. THE STORAGE USED FOR THE TABLE IS MARKED OVERLAYABLE IN A SPECIAL WAY AND THE SPACE IS RELEASED ONLY WHEN IT IS NEEDED. THUS, THE TABLE NEED NEITHER BE

READ FROM DISK FOR EACH MESSAGE PRODUCED NOR REMAIN IN CORE PERMANENTLY.

CHANGE NO. 51 (23 CARDS).

FM OVERRIDE OF SV MESSAGE

THIS CHANGE MAKES AN "FM" MESSAGE OVERRIDE AN "SV" EVEN IN THE CASE WHERE ALL UNITS OF A PARTICULAR TYPE HAVE BEEN SAVED. THE "LP RQD" MESSAGE NOW HAS AN ALTERNATE FORM "LP FM RQD" INDICATING THAT A FORM AND AN UNIT ARE REQUIRED. FORMS MAY ALSO BE SPECIFIED FOR THE PUNCH IN THE SAME WAY.

CHANGE NO. 52 (41 CARDS).

DUMPCORE TAPE PARITY RECOVERY

THIS CHANGE WILL ALLOW THE PROCEDURE DUMPCORE TO ABORT ITSELF IF AN IRRECOVERABLE TAPE PARITY OCCURS WHILE DUMPING MEMORY TO A TAPE. PREVIOUSLY, THE MCP WOULD HANG WITH AN "UNEXP I/O ERROR". A MESSAGE WILL INFORM THE OPERATOR OF THE TERMINATION OF THE DUMP IN THE FORM OF "=DPMT ABORTED, TRY ANOTHER TAPE", SO THAT THE DPMT CAN BE ACCOMPLISHED ON ANOTHER TAPE.

CHANGE NO. 53 (2263 CARDS).

REWRITE OF LIBRARY MAINTENANCE

THIS CHANGE IS A REWRITE OF LIBRARY MAINTENANCE TO INCORPORATE THE USE OF THE "COPY" CONTROL CARD. IT ALSO EXTENDS THE USE OF"EXCEPT" LISTS TO THE "REMOVE" CONTROL CARD. REFER TO ATTACHED DOCUMENTATION.

CHANGE NO. 54 (610 CARDS).

WRITE PARITY REEL SWITCHING

THIS CHANGE IMPLEMENTS THE WRITE PARITY REEL FACILITY IN THE MCP. THIS NEW FEATURE ALLOWS OBJECT JOBS TO RECOVER FROM A FATAL TAPE WRITE PARITY. WHEN A FATAL WRITE PARITY OCCURS, THE LAST TWO BLOCKS SUCCESFULLY WRITTEN ON THE TAPE FILE ARE READ INTO CORE AND THE FILE IS CLOSED OFF AS IF END-OF-REEL HAD OCCURRED. A NEW TAPE UNIT IS OBTAINED, THE TWO BLOCKS READ INTO CORE ARE WRITTEN TO THE NEW TAPE AND THE BLOCK WHICH ORIGINALLY ENCOUNTERED THE FATAL ERROR IS WRITTEN. THE NEW TAPE FILE IS MARKED AS BEING THE NEXT REEL IN A

MULTI-REEL TAPE FILE. THE ONLY OPERATOR INTERVENTION IN THE WRITE PARITY REEL SWITCH FACILITY IS FOR PROVIDING A SCRATCH TAPE TO FURTHER FACILITATE TAPE PARITY HANDLING. A NEW KEYBOARD MESSAGE HAS BEEN ADDED, NAMELY "RC", WHICH CAN BE USED BY THE OPERATOR WHENEVER AN INORDINATE AMOUNT OF WRITE RETRIES ARE OBSERVED ON A PARTICULAR TAPE UNIT. THE FUNCTION OF THIS INPUT MESSAGE IS TO SIMULATE END-OF-REEL ON THE PARTICULAR TAPE UNIT SPECIFIED (E.G. RCMTA) AND ALLOW THE PROGRAM USING THE TAPE TO OBTAIN ANOTHER TAPE AND CONTINUE WRITING ON THE NEXT REEL. THIS FACILITY SHOULD BE QUITE HELPFUL IN REDUCING RERUN DUE TO TAPE PARITY ERRORS.

CHANGE NO. 55 (9 CARDS).

CORRECTIONS TO LIBRARY MAINT

THIS CHANGE CORRECTS MINOR PROBLEMS ASSOCIATED WITH THE NEW LIBRARY MAINTENANCE. IT INCLUDES:

A. A CHECK TO INSURE TAPES ARE SET NOT-IN-USE WHEN LIBMAIN/DISK IS DS-ED.
B. ELIMINATION OF INCORRECT I/O-S WHEN DS-ING.
C. A FIX TO THE NO USER DISK MESSAGE.
D. A CORRECTION TO THE CODE FOR CREATING NEW NAMES ON OUTPUT UNITS WHEN A JOB IS FORKED.

CHANGE NO. 56 (6 CARDS).

CORRECTIONS FOR MARK XVI.0

THIS CHANGE CORRECTS THREE PROBLEMS ASSOCIATED WITH THE NEW RELEASE.

1. INCORRECT HANDLING OF "COPY" SPECIFICATIONS ON THE FILE LABEL EQUATION CARD.
2. FILES LEFT IN-USE IF SPECIFIED MORE THAN ONCE IN THE NAME LISTS OF A LIBRARY MAINTENANCE CONTROL CARD.
3. INCORRECT HANDLING OF THE NUMBER OF COPIES SPECIFIED USING LABEL EQUATION ON PIGGY-BACKED PRINTER BACK UP TAPES. THE CORRECT NUMBER OF COPIES MAY BE OBTAINED FROM PBT-S CREATED PRIOR TO THIS RELEASE BY USING THE "PB" KEYBOARD REQUEST ACCORDING TO THE DOCUMENTATION OF THE MARK XV.2.0 SYSTEM RELEASE (APPENDIX F). DATED 12/15/73.

CHANGE NO. 57 (6 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

TSSMCP CHANGES 1 THROUGH 75.

CHANGE NO. 1 (2 CARDS).

CANDE FILE BLOCKING

THIS CHANGE ALTERS MCP PROCEDURE INDIANBUY TO CREATE CANDE FILES WITH STANDARD BLOCKING WHEN REQUESTED BY CANDE. THIS AVOIDS PROBLEMS WHICH OCCUR WHEN A FILE IS CREATED AND THEN SAVED WITHOUT ENTERING ANY RECORDS INTO THE FILE.

CHANGE NO. 2 (20 CARDS).

FIXED BIT IN HEADER, TABS IN-USE

THIS CHANGE ALLOWS CANDE TO REQUEST THAT A FILE BE CREATED WITH THE "FIXED" (HEADER[4].[42:1]) BIT SET SO THAT THE FILE WILL NOT BE MOVED WHEN AN "SQ" REQUEST IS ENTERED. THE "FIXED" BIT IS ALSO SET IN THE "TANK/DISK" FILE WHEN CANDE IS INITIALIZED TO PREVENT THIS FILE FROM BEING MOVED WHEN CANDE IS NOT RUNNING. THESE CHANGES ARE NECESSARY BECAUSE CANDE AND ITS SATELLITE ROUTINES SOMETIMES EXECUTE I/O-S TO ACTUAL DISK ADDRESSES INSTEAD OF RELATIVE DISK ADDRESSES. THIS CHANGE ALSO ALTERS INDIANBUY SO THAT A REMOVE FILE REQUEST ON A FILE WHICH IS MARKED IN-USE WILL RETURN A FLAGGED RESULT TO CANDE, INDICATING THE IN-USE CONDITION. CANDE WILL NOW CLEAR THE TAB FILES (THE 1P AND 1T FILES) WHEN IT ATTEMPTS TO CREATE NEW TAB FILES WHILE THE OLD FILES ARE MARKED IN-USE.

NOTE CHANGE CANDE XV.3.01 IS REQUIRED FOR PROPER OPERATION OF THIS CHANGE.

CHANGE NO. 3 (1 CARD).

NUM OF LINES IN QT MSG

SEE MCP CHANGE NUMBER XV.3.02

CHANGE NO. 4 (1 CARD).

B6500 LOAD OF A B5500 TAPE

SEE MCP CHANGE NUMBER XV.3.03

CHANGE NO. 5 (9 CARDS).

QMK STATUS DURING SELECTION

THIS CHANGE PREVENTS THE MCP FROM ATTEMPTING TO ACCESS A JOB'S PRT WHEN THE JOB IS STILL IN THE SELECTION PROCESS. THIS AVOIDS TRYING TO ACCESS THE PRT BEFORE IT HAS BEEN INITIALIZED. THIS CHANGE ALSO CHECKS THE RELATIVE POSITION OF A JOB'S "PRT" WITHIN ITS CORE AREA TO DETERMINE WHETHER THE JOB IS "EXPANDED" BUT NOT YET SWAPPED BACK INTO THE MIX. ACCESS TO THE PRT IS DISALLOWED SINCE THE RELATIVE POSITION OF THE PRT ON DISK IS NOT READILY AVAILABLE TO THE MCP AT THAT POINT IN TIME.

CHANGE NO. 6 (1 CARD).

REMOVE NAME, OPTN 2

SEE MCP CHANGE NUMBER XV.3.04

CHANGE NO. 7 (1 CARD).

SCHED LINES, HDR SPACE

THIS CHANGE CORRECTS AN ERROR WHICH COULD LEAVE 30 WORDS OF SPACE IN USE IN THE UNLIKELY EVENT THAT WHEN A NEW SCHEDULE INPUT FILE IS BEING LINKED INTO THE TASK QUEUE A FILE WITH THE SAME NAME ALREADY EXISTS ON DISK.

CHANGE NO. 8 (7 CARDS).

USE OF NT1 IN CONTROLCARD

SEE MCP CHANGE NUMBER XV.3.05

CHANGE NO. 9 (27 CARDS).

PACKET = ZIPARRAY

SEE MCP CHANGE NUMBER XV.3.06

CHANGE NO. 10 (4 CARDS).

NEWLOGGING - 2 PROCESSOR

SEE MCP CHANGE NUMBER XV.3.07

CHANGE NO. 11 (1 CARD).

STARTIMING - SHAREDISK

SEE MCP CHANGE NUMBER XV.3.08

CHANGE NO. 12 (80 CARDS).

NEWLOGGING - WM RESPONSE

SEE MCP CHANGE NUMBER XV.3.09

CHANGE NO. 13 (1 CARD).

INDIANBOY & GIRL INITIATION

THIS CHANGE INSURES THAT A CALL ON COMM15 (THE INDEPENDENT STARTER FOR CANDE) WILL RESULT IN A DIRECT RETURN TO THE OUTER BLOCK.

CHANGE NO. 14 (102 CARDS).

GETSPACE IMPROVEMENTS

THE PRIMARY FUNCTION OF THIS CHANGE IS TO IMPROVE COMPATIBILITY WITH THE BATCH MCP. TO THIS END, THE FOLLOWING HAVE BEEN DONE:

1. THE NOMEM FIELD IN TOGGLE IS MOVED TO 9:3 .
2. THE WAY IN WHICH LIBRARY MAINTENANCE OBTAINS SPACE IS REORGANIZED AND MADE MORE EFFICIENT.
3. THE INTERPRETATION OF THE FINAL PARAMETER TO PROCEDURE GETSPACE HAS BEEN MODIFIED SO THAT A VALUE OF 0 IS NORMAL AND 64 IS USED TO SPECIFY SPACE WHICH SHOULD NOT BE OBTAINED AS SAVE SPACE DURING THE PERIOD AFTER A NO-MEM. ALSO, THIS CHANGE ENSURES THAT NO-MEMS OCCURRING BELOW THE FENCE DO NOT AFFECT THE WAY SPACE IS ALLOCATED ABOVE THE FENCE.

CHANGE NO. 15 (7 CARDS).

FILE-IN-USE ERRORS

SEE MCP CHANGE NUMBER XV.3.11

CHANGE NO. 16 (5 CARDS).

HDR SPACE LIBRARYZERO

SEE MCP CHANGE NUMBER XV.3.12

CHANGE NO. 17 (3 CARDS).

RRRMECH IN TAPE RETRY ON LABEL

THIS CHANGE CORRECTS AN ERROR IN THE INTERLOCKING BETWEEN THE MCP PROCEDURE STATUS AND OTHER PROCEDURES AND PROGRAMS WHICH ACCESS TAPE UNITS. THE ERROR OCCURRED IF A PARITY RETRY WERE PERFORMED ON AN I/O TO THE TAPE LABEL. USUALLY A SYSTEM HANG RESULTED. THE ERROR WAS INTRODUCED IN PATCH XV.2.19 .

CHANGE NO. 18 (47 CARDS).

NEWLOGGING CORRECTIONS

THIS CHANGE CORRECTS SEVERAL ERRORS IN THE OUTPUT OF ACCUMULATED PROCESSOR TIME FOR A JOB. THESE ERRORS OCCURRED BECAUSE OF INCORRECT TESTS BEING MADE UNDER THE NEWLOGGING OPTION IN THE PROCEDURES INDIANBOY, TIMEUSED AND WHATSGOINGON. THIS CHANGE ALSO MAKES THE NEWLOGGING CODE IN SELECTRUN COMPATIBLE WITH THE DCMCP.

CHANGE NO. 19 (5 CARDS).

CCSET HEADER SPACE RELEASE

SEE MCP CHANGE NUMBER XV.3.16

CHANGE NO. 20 (1 CARD).

TANK FILE HDR ADDRS TO CANDE

THIS CHANGE CORRECTS AN ERROR IN WHICH THE TANK FILE HEADER ADDRESS WAS NOT PASSED TO CANDE ON A RESTART.

CHANGE NO. 21 (12 CARDS).

AS AND QS REPLACE SM AND RS

THIS CHANGE CHANGES THE MNEMONICS FOR THE SM AND RS KEYBORD REQUESTS TO AS (ACTIVITY SUMMARY) AND QS (QUICK SS) RESPECTIVELY. THIS IS DONE TO ELIMINATE CONFUSION WITH THE USAGE OF SM AND RS IN THE DATACOM MCP.

CHANGE NO. 22 (29 CARDS).

AUTOMATIC OK SUPPRESSION

THIS CHANGE CORRECTS CONDITIONS IN WHICH JOBS WAITING FOR OPERATOR INTERVENTION WERE INAPPROPRIATELY OK-ED BY THE MCP. IN PARTICULAR, THIS ERROR COULD OCCUR WHILE WAITING FOR AN FM AND AFTER A DISK PARITY DURING PRINTING OF A BACK UP FILE. IN THOSE CASES, THE ERROR CAUSED PRINTING TO BE STARTED WITHOUT OPERATOR ACTION.

CHANGE NO. 23 (4 CARDS).

MTA ACCESS AFTER DS DURING FM

SEE MCP CHANGE NUMBER XV.3.17

CHANGE NO. 24 (33 CARDS).

REASON ADDED TO SCHEDULED MSG

THIS CHANGE CAUSES A MESSAGE TO BE SPOUTED IN ALL CASES IF A JOB CANNOT BE STARTED AFTER A REQUEST TO XS OR ES IT. THE FORMAT OF THIS MESSAGE IS:

<PRIORITY>:<JOB SPECIFIER> NOT XS-ED, <REASON>

THIS REPLACES THE MESSAGE "NOT XS-ED (<REASON>)" WHICH WAS USED PREVIOUSLY. IN ADDITION, THE <REASON> FOR WHICH A JOB IS INITIALLY SCHEDULED HAS BEEN ADDED TO THE SCHEDULED MESSAGE WHICH NOW HAS THE FOLLOWING

<PRIORITY>:<JOB SPECIFIER> SCHEDULED <TIME>, <REASON>

THE <REASONS> FOR WHICH A JOB IS SCHEDULED ARE:

NO MEM	AS DETERMINED BY THE CORE ESTIMATE AND THE MULTI-PROCESSING FACTOR.
NO SWAP DISK	
TOO MANY JOBS	MIXMAX JOBS ALREADY RUNNING
BACKGROUND	OTHER BACKGROUND JOBS ARE ALREADY RUNNING

CHANGE NO. 25 (2 CARDS).

UNIT FIELD IN TAPE RETRY MSG

THIS RESTORES THE CORRECT UNIT DESIGNATE TO THE RESULT DESCRIPTOR OF THE TAPE FAILURE HARDWARE ERROR MESSAGE WHICH IS PLACED IN THE LOG.

CHANGE NO. 26 (2 CARDS).

SYNTAX ERR, PACKETS VS AUXMEM

SEE MCP CHANGE NUMBER XV.3.20

CHANGE NO. 27 (5 CARDS).

PBD FILES, NUM> 8000, COPIES

SEE MCP CHANGE NUMBER XV.2.21

CHANGE NO. 28 (47 CARDS).

INV LINK TEST IN SWAPINGIO

THIS CHANGE IMPROVES THE CHANCES OF SYSTEM RECOVERY AFTER AN INVALID MEMORY LINK ABOVE THE FENCE BY INSTITUTING A TEST FOR THIS DURING SWAPPING. IF AN INVALID LINK IS DETECTED, AN ATTEMPT IS MADE TO DS THE JOB. PREVIOUSLY, THE SYSTEM WENT INTO A LOOP.

CHANGE NO. 29 (1 CARD).

INCORRECT PUNT

SEE MCP CHANGE NUMBER XV.3.22

CHANGE NO. 30 (35 CARDS).

OPRTR ACTION FOR RUN CNTRL CRD

WITH THIS CHANGE, IF A RUN CONTROL CARD IS ENTERED, THE SYSTEM WILL PRINT:

*RUN CONTROL CARD, MIX=0:<CONTROL CARD>

AT THE SPO AND THEN SUSPEND PROCESSING OF THAT CARD PENDING OPERATOR INTERVENTION. THE OPERATOR MAY ENTER EITHER

OOK, ALLOWING THE JOB TO RUN

OR

OQT, CAUSING A CONTROL CARD ERROR TO BE GIVEN.

CHANGE NO. 31 (4 CARDS).

RRRMECH ERR PB KBD REQUEST

SEE MCP CHANGE NUMBER XV.3.26

CHANGE NO. 32 (2 CARDS).

AS KBD REQUEST, WAITING JOBS

THIS CHANGE CORRECTS AN ERROR IN THE PROCEDURE WHICH HANDLES THE AS KEYBOARD REQUEST WHICH COULD CAUSE THE INFORMATION CONCERNING WHY A JOB IS WAITING TO BE PRINTED OUT OF ORDER.

CHANGE NO. 33 (3 CARDS).

WORDS RQD ADDED TO SCHEDULE MSG

THIS CHANGE ALTERS THE SCHEDULE MESSAGE TO INCLUDE THE NUMBER OF WORDS REQUIRED WHEN A JOB IS SCHEDULED DUE TO A NO MEM CONDITION. THE FORMAT FOR THAT CASE IS NOW:

<PRIORITY>:<JOB SPECIFIER>SCHEDULED<TIME>,NEEDS<NUMBER OF WORDS>

CHANGE NO. 34 (23 CARDS).

FINDINPUT WAITS FOR CONTROLCRD

SEE MCP CHANGE NUMBER XV.3.029

CHANGE NO. 35 (32 CARDS).

MONITOR INDEPENDENT OF AUXMEM

THIS CHANGE MAKES THE MONITOR AND AUXMEM COMPILE-TIME OPTIONS INDEPENDENT OF EACH OTHER. THAT IS, AN MCP CAN BE COMPILED WITH MONITOR SET AND AUXMEM RESET.

CHANGE NO. 36 (33 CARDS).

PBD, TWO FILES TO ONE PRINTER

SEE MCP CHANGE NUMBER XV.3.31

CHANGE NO. 37 (19 CARDS).

CANDE & MCP OUTPUT AFTER BREAK

THIS CHANGE ALLOWS CANDE AND THE MCP TO SEND MESSAGES TO A REMOTE STATION ON WHICH A BREAK ON OUTPUT HAS OCCURRED. OUTPUT STILL WILL NOT BE ALLOWED FROM AN USER PROGRAM UNTIL IT DOES A READ OR GOES TO EOJ. THE PURPOSE OF THIS CHANGE IS TO ALLOW MESSAGES FROM THE OPERATOR AND RESPONSES TO QUESTION MARK COMMANDS TO GO TO THE USER AFTER A BREAK. THE CHANGE ALSO ELIMINATES AN UNNECESSARY SWAP WHICH OCCURRED IF A BREAK WERE DONE ON TANKED OUTPUT WHILE THE JOB WAS WAITING FOR INPUT.

CHANGE NO. 38 (1 CARD).

PBT REEL SWITCH HANG

THIS CHANGE CORRECTS A SYSTEM HANG WHICH HAPPENED WHEN A REEL SWITCH OCCURRED WHILE PRINTING A PRINTER BACK UP TAPE.

CHANGE NO. 39 (3 CARDS).

AUTO-RM OF FILE BEING REMOVED

THIS CHANGE CORRECTS AN ERROR WHICH OCCURRED IF, DURING AN AUTOMATIC RM, AN OLDER VERSION OF THE FILE ALREADY ON DISK WAS CONCURRENTLY IN THE PROCESS OF BEING REMOVED. IN THAT CASE, IT WAS POSSIBLE THAT THE NEW VERSION WAS NOT ENTERED INTO THE DIRECTORY AFTER THE OLD ONE WAS REMOVED.

CHANGE NO. 40 (21 CARDS).

DELAY FUNCTION ADDED

THIS CHANGE ADDS THE ALGOL DELAY FUNCTION TO TIMESHARING, NOTE THAT IF MORE THAN 15 SECONDS ELAPSE BEFORE THE CONDITION IS SATISFIED THE JOB WIL BE DS-ED FOR AN "INVALD SLEEP".

CHANGE NO. 41 (4 CARDS).

H/L INTRINSIC INITIALIZATION

SEE MCP CHANGE NUMBER XV.3.32

CHANGE NO. 42 (20 CARDS).

INVLD AUXMEM IO MESSAGE

SEE MCP CHANGE NUMBER XV.3.34

CHANGE NO. 43 (24 CARDS).

MIX INDEX FOR QM STATUS

THIS CHANGE CORRECTS AN ERROR WHICH OCCURRED WHEN TWO OR MORE TERMINALS, LOGGED IN UNDER THE SAME USERCODE, EXECUTED THE SAME PROGRAM. IN THAT CASE, IF A QM STATUS WERE DONE, IT ALWAYS RETURNED INFORMATION PERTAINING TO THE COPY OF THE JOB WHICH HAD THE LOWEST MIX INDEX (QM = QUESTION MARK).

CHANGE NO. 44 (4 CARDS).

AUTORM CREATE TIME, SAVEFAC<7

THIS CHANGE CORRECTS TWO PROBLEMS:

1. IF AN ATTEMPT WERE MADE TO USE A SAVE FACTOR OF LESS THAN 7 IT WAS SET TO 7. NOW ANY NON-ZERO VALUE MAY BE USED.
2. WHEN AN AUTOMATIC RM WAS DONE, THE CREATION TIME WAS NOT UPDATED IN THE HEADER. THIS CAUSED THE SECOND VERSION OF THE FILE TO BE INCORRECTLY LOGGED WHEN IT WAS SUBSEQUENTLY REMOVED AND MADE IT IMPOSSIBLE TO USE THE CANDE "WHATS" COMMAND TO DETERMINE WHICH VERSION OF A FILE WAS ACTUALLY ON DISK.

CHANGE NO. 45 (133 CARDS).

MAX SLEEP OF 15 SECS

THIS CHANGE CAUSES A JOB RUNNING ABOVE THE FENCE WHICH SLEEPS IN CORE FOR MORE THAN 15 SECONDS TO BE DS-ED WITH THE EXPLANATION:

INVALD SLEEP

THIS CHANGE SHOULD CORRECT SOME OF THE OCCASIONS WHEN THE MCP CEASES TO SWAP JOBS.

CHANGE NO. 46 (5 CARDS).

EMBRACE, WRITE PARITY-UNHOOQUE

THIS CHANGE CORRECTS A DEADLY EMBRACE THAT COULD OCCUR BETWEEN THE MCP PROCEDURES UNHOOQUE, TAPEPARITYRETRY AND MAINTLOGGER. IT OCCURRED ONLY AFTER A WRITE RETRY FAILED AND THE STOPTEST OPTION WAS RESET.

CHANGE NO. 47 (1 CARD).

INTRINSIC ABOVE & BELOW FENCE

THIS CHANGE CORRECTS AN ERROR WHICH CAUSED AN EXCESSIVE AMOUNT OF SPACE TO BE OBTAINED FOR AN INTRINSIC ABOVE THE FENCE IF A COPY OF THE INTRINSIC WAS ALSO PRESENT BELOW THE FENCE. FREQUENTLY THIS ERROR CAUSED JOBS TO BE DS-ED FOR BEING "OUT OF MEM".

CHANGE NO. 48 (4 CARDS).

EXPECTED IO ERRS

SEE MCP CHANGE NUMBER XV.3.37

CHANGE NO. 49 (18 CARDS).

MULTIPOINT ERRORS

THIS CHANGE CORRECTS THE FOLLOWING ERRORS IN THE HANDLING OF MULTIPOINT DATACOM LINES:

1. IF AN UNRECOGNIZED ADDRESS WAS RECEIVED, THE SYSTEM HUNG.
2. AFTER AN UNRECOVERED PARITY ERROR, THE LINE WAS NOT PROPERLY DISCONNECTED.

THE PATCH ALSO OPTIMIZES SOME CODE IN THE AREA WHICH HANDLES ACKS TO A MESSAGE.

CHANGE NO. 50 (17 CARDS).

BREAK/WRU TRAN NUM ERROR

THIS CHANGE ELIMINATES A TRANSMISSION NUMBER ERROR THAT SOMETIMES OCCURRED AT A TC500 AFTER A "<QM>BREAK" OR "<QM>WRU" WAS ENTERED (QM =QUESTION MARK).

CHANGE NO. 51 (1 CARD).

DUP ON MESSAGE RESTORED

THIS CHANGE CAUSES THE MESSAGES WHICH INDICATED THE UNITS ON WHICH DUP FILES RESIDE TO BE PROPERLY PRINTED AT THE CONSOLE. PREVIOUSLY, THEY WERE LEFT IN CORE, POSSIBLY LEADING TO EVENTUAL NO MEM SITUATIONS.

CHANGE NO. 52 (1 CARD).

NULL WY

THIS CHANGE CORRECTLY POSITIONS THE NULL WY MESSAGE WHEN REPRINTED AT THE CONSOLE IN RESPONSE TO A WY KEYBOARD REQUEST.

CHANGE NO. 53 (2 CARDS).

PUNT FOR ESPDISK ERROR

SEE MCP CHANGE NUMBER XV.3.40

CHANGE NO. 54 (1 CARD).

SYSTEM NUMBER IN SEARCH STMT

SEE MCP CHANGE NUMBER XV.3.41

CHANGE NO. 55 (14 CARDS).

SHAREDISK SOSTOP

SEE MCP CHANGE NUMBER XV.3.42

CHANGE NO. 56 (2 CARDS).

DS JOB WITH BLOCK OVER 1890 WD

SEE MCP CHANGE NUMBER XV.3.43

CHANGE NO. 57 (13 CARDS).

RN# ACCEPTS LESS THAN 4 DIGITS

SEE MCP CHANGE NUMBER XV.3.44

CHANGE NO. 58 (4 CARDS).

JULIAN DATE IN WD AND H/L

SEE MCP CHANGE NUMBER XV.3.45

CHANGE NO. 59 (6 CARDS).

SCHEDULE LINE BUGS

THIS CHANGE CORRECTS THE FOLLOWING ERRORS IN THE PROCESSING OF SCHEDULE LINES.

1. OUTPUT OF EXACTLY 64 CHARACTERS WAS LOST
2. A GROUPMARK WAS PUT IN THE OUTPUT BEHIND THE QUESTION MARK INDICATING THAT A PROGRAM ASKED FOR INPUT. WHEN LISTED AT A TERMINAL, THIS APPEARED AS TWO QUESTION MARKS.
3. LINES WHICH WERE SCHEDULED USING THE AFTER FACILITY WERE NOT CORRECTLY RUN IF A IR WERE DONE OR AFTER MIDNIGHT OCCURRED.

CHANGE NO. 60 (12 CARDS).

PAUSE FOR DP AFTER CANDE ERR

THIS CHANGE CORRECTS AN ERROR WHICH CAUSED THE MCP TO READY THE SPO FOR A POSSIBLE DP TWICE AFTER A CANDE ERROR IF THE TERMNATE OPTION WERE RESET. NOW ABNORMAL CANDE TERMINATIONS ARE TREATED AS IF TERMNATE WERE RESET, THAT IS, THE MESSAGE --CANDE ERROR, PLEASE TAKE DUMP ETC IS NO LONGER OUTPUT. INSTEAD, THE APPROPRIATE ERROR MESSAGE IS PRINTED AND THE SPO IS READIED. ALSO, WHEN A JOB IS BEING ES-ED, THE MCP WILL NO LONGER STOP FOR A DP. SIMILARLY, IF CANDE IS DS-ED BY THE OPERATOR IT WILL GO DIRECTLY TO END-OF-JOB. THEREFORE, IF A DP IS DESIRED, IT MUST BE DONE BEFORE CANDE IS DS-ED.

CHANGE NO. 61 (3 CARDS).

HEADER AND LEVEL CARDS

THIS CHANGE UPDATES THE HEADER AND LEVEL CARDS.

CHANGE NO. 62 (47 CARDS).

SYSTEM JOB CONTROL

SEE MCP CHANGE NUMBER XV.3.48

CHANGE NO. 63 (12 CARDS).

LOCALIZE CONTROLCARD VARIABLES

SEE MCP CHANGE NUMBER XV.3.49

CHANGE NO. 64 (69 CARDS).

OVERLAY LIBMSG TABLE

SEE MCP CHANGE NUMBER XV.3.50

CHANGE NO. 65 (25 CARDS).

FM OVERRIDE OF SV MESSAGE

SEE MCP CHANGE NUMBER XV.3.51

CHANGE NO. 66 (41 CARDS).

DUMPCORE TAPE PARITY RECOVERY

SEE MCP CHANGE NUMBER XV.3.52

CHANGE NO. 67 (2322 CARDS).

REWRITE OF LIBRARY MAINTENANCE

SEE MCP CHANGE NUMBER XV.3.53

CHANGE NO. 68 (608 CARDS).

WRITE PARITY REEL SWITCHING

SEE MCP CHANGE NUMBER XV.3.54

CHANGE NO. 69 (15 CARDS).

CORRECTIONS TO LIBRARY MAINT

THIS CHANGE CORRECTS MINOR PROBLEMS ASSOCIATED WITH THE NEW LIBRARY MAINTENANCE. IT INCLUDES:

A. A CHECK TO INSURE TAPES ARE SET NOT-IN-USE WHEN LIBMAIN/DISK IS DS-ED.

B. ELIMINATION OF INCORRECT I/O-S WHEN DS-ING.

C. A FIX TO THE NO USER DISK MESSAGE.

D. A CORRECTION TO THE CODE FOR CREATING NEW NAMES ON OUTPUT UNITS WHEN A JOB IS FORKED.

CHANGE NO. 70 (11 CARDS).

CORRECTION ON INVALID SLEEP

THIS CHANGE CORRECTS A LOGGING PROBLEM WHICH OCCURRED WITH NEWLOGGING NOT SET. IN ADDITION, THE PROBABILITY OF AN INVALID SLEEP HAPPENING IS REDUCED BY MAKING THE TIME LIMIT DEPENDENT ON THE NUMBER OF PROGRAMS IN THE MIX.

CHANGE NO. 71 (6 CARDS).

CORRECTIONS FOR MARK XVI.0

SEE MCP CHANGE NUMBER XV.3.56

CHANGE NO. 72 (5 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

INTRINSICS CHANGES 1 THROUGH 6.

CHANGE NO. 1 (5 CARDS).

COBOL ATT = TYPE CHECK

THIS CHANGE CORRECTS AN ERROR IN THE CHECKING FOR AN UNALTERABLE FILE TYPE, NAMELY TYPE 26, DISK PROTECT. THIS IS ALSO AN UNALTERABLE FILE TYPE.

CHANGE NO. 2 (5 CARDS).

SORT OUTPUT ARRAY ACCESSING

THIS CHANGE DOES TWO THINGS.

1. IF AN ATTEMPT IS MADE TO ACCESS THE ARRAY PASSED TO THE OUTPUT PROCEDURE OF AN ALGOL SORT OR MERGE WHEN THE OTHER PARAMETER IS TRUE (INDICATING THAT THE ARRAY IS NOT VALID), THE JOB WILL BE DS-ED FOR INVALID ADDRESS. PREVIOUSLY, THE JOB CONTINUED, AND COULD CAUSE SYSTEM HANGS, DEPENDING ON HOW THE ARRAY WAS ACCESSED.
2. THE MESSAGE INDICATING THAT THE I/O ERROR HAS OCCURRED IN AN ALGOL SORT HAS BEEN CORRECTED. PREVIOUSLY, GARBAGE WAS PRINTED AND THE SYSTEM USUALLY HUNG.

CHANGE NO. 3 (18 CARDS).

COBOL REEL SWITCH USE ROUTINES

THIS CHANGE INHIBITS THE CALLING OF COBOL REEL SWITCH USE ROUTINES WHEN A REEL SWITCH IS DETECTED BECAUSE OF A REEL SWITCH PARITY, AS EXPLAINED IN PATCH MCP.XV.3.54.THIS IS REQUIRED TO PREVENT THE COBOL PROGRAMS FROM LOOKING AT BEGINNING AND ENDING LABELS WHICH HAVE BEEN CREATED BY THE MCP IN ORDER TO RECOVER FROM AN OTHERWISE IRRECOVERABLE WRITE PARITY ON TAPE.

CHANGE NO. 4 (3 CARDS).

HEADER AND LEVEL CARDS

THIS CHANGE UPDATES THE HEADER AND LEVEL CARDS.

CHANGE NO. 5 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

CHANGE NO. 6 (1 CARD).

DISK BLOCKING OVER 930 WORDS

THIS CHANGE CORRECTS A PROBLEM CAUSED BY A PREVIOUS PATCH (XV.2.03) WHICH PREVENTED WRITING TO DISK IN BLOCKS OF GREATER THAN 930 WORDS.

ALGOL CHANGES 1 THROUGH 6.

CHANGE NO. 1 (3 CARDS).

HEADER AND LEVEL CARDS

THIS CHANGE UPDATES THE HEADER AND LEVEL CARDS.

CHANGE NO. 2 (1 CARD).

SEGMENT SIZE 4092 SYLLABLES

THIS CHANGE WILL PREVENT THE COMPILER FROM HANGING THE SYSTEM WHEN IT COMPILES A PROGRAM WITH A SEGMENT SIZE OF 4093 SYLLABLES OR GREATER. THE COMPILER WILL LMIT THE SYNTAX ERROR NUMBER 200, SEGMENT TOO LARGE (>4093 SYLLABLES).

CHANGE NO. 3 (2 CARDS).

DOLLAR CARD AFTER MAKCAST CALL

THIS CHANGE WILL NOW ALLOW THE COMPILER TO ACCEPT A DOLLAR CARD IMMEDIATELY FOLLOWING A MAKCAST CALL.

CHANGE NO. 4 (9 CARDS).

SEQUENCE ERROR COUNTER

THIS CHANGE WILL COUNT THE NUMBER OF SEQUENCE ERRORS AND PRINT THE TOTAL AT THE END OF THE LISTING IF THE "CHECK" OPTION IS SET.

CHANGE NO. 5 (7 CARDS).

SEQUENCE AND SYNTAX ERROR

THIS CHANGE MAKES THE LISTING OF SEQUENCE AND SYNTAX ERRORS MORE VISIBLE ESPECIALLY IF $BEND HAS BEEN SET.

CHANGE NO. 6 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

BASIC CHANGES 1 AND 2.

CHANGE NO. 1 (1 CARD).

HEADER CARD

THIS CHANGE UPDATES THE HEADER CARD.

CHANGE NO. 2 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

COBOL CHANGES 1 AND 2.

CHANGE NO. 1 (2 CARDS).

HEADER AND LEVEL CARDS

THIS CHANGE UPDATES THE HEADER AND LEVEL CARDS.

CHANGE NO. 2 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

COBOL68 CHANGES 1 THROUGH 3.

CHANGE NO. 1 (4 CARDS).

TIME(-1) FUNCTION

THIS CHANGE PREVENTS AN INCORRECT SYNTAX ERROR FROM BEING PRODUCED WHEN A SIGNED NUMERIC LITERAL WAS USED AS THE ARGUMENT TO THE TIME FUNCTION, SUCH AS TIME(-1).

CHANGE NO. 2 (2 CARDS).

LEVEL CARDS

THIS CHANGE UPDATES THE LEVEL CARDS.

CHANGE NO. 3 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

ESPOL CHANGES 1 AND 2.

~~CHANGE NO. 1 (3 CARDS).~~

~~HEADER AND LEVEL CARDS~~
THIS CHANGE UPDATES THE HEADER AND LEVEL CARDS.

CHANGE NO. 2 (4 CARDS).

COPY RIGHT NOTICE
THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

FORTRAN CHANGES 1 AND 2.

CHANGE NO. 1 (3 CARDS).

HEADER AND LEVEL CARDS

THIS CHANGE UPDATES THE HEADER AND LEVEL CARDS.

CHANGE NO. 2 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

XALGOL CHANGES 1 THROUGH 6.

CHANGE NO. 1 (3 CARDS).

HEADER AND LEVEL CARDS

THIS CHANGE UPDATES THE HEADER AND LEVEL CARDS.

CHANGE NO. 2 (2 CARDS).

DOLLAR CARD AFTER MAKCAST CALL

SEE ALGOL CHANGE NUMBER XV.3.03

CHANGE NO. 3 (1 CARD).

SEGMENT SIZE 4092 SYLLABLES

SEE ALGOL CHANGE NUMBER XV.3.02

CHANGE NO. 4 (9 CARDS).

SEQUENCE ERROR COUNTER

SEE ALGOL CHANGE NUMBER XV.3.04

CHANGE NO. 5 (7 CARDS).

SEQUENCE AND SYNTAX ERROR

SEE ALGOL CHANGE NUMBER XV.3.05

CHANGE NO. 6 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

CANDE CHANGES 1 THROUGH 4.

CHANGE NO. 1 (103 CARDS).

FIXED BIT IN HEADER, TABS IN-USE

THIS CHANGE ALLOWS CANDE/TSHARER TO RECOGNIZE AN IN-USE CONDITION ON FILES WHICH IT IS ATTEMPTING TO CREATE, WHEN CREATING NEW TAB FILES (THE 1P AND THE 1T FILES). PREVIOUSLY CREATED FILES WHICH ARE FLAGGED BY THE MCP AS IN-USE ARE NOW CLEARED SO THAT PROCESSING MAY CONTINUE. 1T ALSO CAUSES THE "1P" FILE TO BE FIXED FOR SQUASHING (SQ).

NOTE CHANGE TSSMCP XV.3.03 IS REQUIRED FOR PROPER OPERATION OF THIS CHANGE.

CHANGE NO. 2 (1 CARD).

SAVE WKFIL NAME ON DSK AFT.LOD

THIS CHANGE CORRECTS AN ERROR IN WHICH THE NAME OF THE WORKFILE WAS NOT SAVED ON DISK AFTER A FILE WAS LOADED. IF CANDE WAS DS-ED BEFORE RECORDS WERE ALTERED AND THEN STARTED AGAIN, THE WORKFILE WOULD BE INCORRECT.

CHANGE NO. 3 (4 CARDS).

SAVE FACTOR UPDATE

PREVIOUSLY, WHEN A FILE WHICH HAD BEEN LOADED WAS SAVED, THE SAVE FACTOR WAS SET TO 7 UNLESS A VALUE WAS INCLUDED IN THE SAVE COMMAND. NOW THE FILE WILL KEEP THE SAVE FACTOR IT HAD BEFORE IT WAS LOADED.

CHANGE NO. 4 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

APPEND CHANGE 1.

~~CHANGE NO. 1 (4 CARDS).~~

~~COPY RIGHT NOTICE~~
THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

COPY CHANGE 1.

~~CHANGE NO. 1 (4 CARDS).~~

~~COPY RIGHT NOTICE~~

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

DELETE CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

FIND CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

GUARD CHANGE 1.

~~CHANGE NO. 1 (4 CARDS).~~

~~COPY RIGHT NOTICE~~
THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

HARD CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

HELP CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

LFILES CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

LIST CHANGES 1 AND 2.

CHANGE NO. 1 (5 CARDS).

INVALID LINK LIST/CANDE

THIS CHANGE PREVENTS THE LIST/CANDE PROGRAM FROM TERMINATING WITH AN INVALID LINK WHEN IT PROCESSES RECORDS FROM A TANK/DISK FILE THAT HAS BEEN PARTIALLY DESTROYED (OVERWRITTEN). SINCE THE SIZE OF THE USERS INPUT RECORD IS CONTAINED IN THE TANK/DISK FILE, IT WAS PREVIOUSLY POSSIBLE FOR THE LIST/CANDE PROGRAM TO CALCULATE A MEMORY ADDRESS WHICH WAS OUTSIDE OF THE ARRAY USED TO PROCESS THE INFORMATION.

CHANGE NO. 2 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

LOAD CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

MERG CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

PAPER CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

PUNCH CHANGES 1 AND 2.

CHANGE NO. 1 (1 CARD).

FILENAME IN TAPE LEADER

THIS CHANGE INSURES THAT THE CORRECT FILE NAME WILL BE PUNCHED INTO THE LEADER OF A PAPER TAPE.

CHANGE NO. 2 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

QUIKLST CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

REPLACE CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

RESEQ CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE.

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

RESEQB CHANGE 1.

~~CHANGE NO. 1 (4 CARDS).~~

~~COPY RIGHT NOTICE~~

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

SCHEDUL CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

UPDATE CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

USER CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

PMERGE CHANGES 1 THROUGH 8.

CHANGE NO. 1 (162 CARDS).

NEW OPTION - ZIPARRAY

THIS CHANGE IMPLEMENTS A NEW $ OPTION IN PATCH/MERGE, NAMELY "ZIPARRAY". THIS OPTION WHEN SET WILL CAUSE PATCH/MERGE TO DO ZIP WITH ARRAY INSTEAD OF THE ZIP WITH FILE. THIS CAN BE USEFUL WHEN RUNNING A PACKET JOB SO THAT THE INFORMATION ABOUT THE ZIPPED JOB WILL APPEAR WITH THE PATCH/MERGE INFORMATION ON THE PACKET PAGE. THIS CHANGE ALSO CORRECTS TWO PROBLEMS. FIRST THE "DELETE"OPTION WAS INCORRECT IN THAT IF MORE THAN ONE DELETE CARD WAS ENCOUNTERED THE PATCH NUMBERS SPECIFIED ON PREVIOUS CARDS WOULD BE OVERWRITTEN BY THE CURRENT CARD RESULTING IN NOT DELETING ALL THE PATCHES SPECIFIED. ALSO ONLY 20 PATCH NUMBERS MAXIMUM WERE ALLOWED TO BE DELETED. THIS NUMBER HAS BEEN INCREASED TO 50 AND MAY BE CHANGED BY A SIMPLE DEFINE CALLED "MAXDEL". SECONDLY, WITH THE OPTION LIST OR LISTI RESET AND AN ERROR OCCURRING IN READING THE INPUT CARD DECK, ONLY THE ERROR MESSAGE WAS LISTED OMITTING THE CARD IMAGE CAUSING THE ERROR. NOW BOTH THE CARD IMAGE AND THE ERROR MESSAGE WILL APPEAR WHEN LIST OR LISTI IS RESET.

CHANGE NO. 2 (31 CARDS).

LABEL EQUATION OF FILES

THIS CHANGE ALLOWS LABEL EQUATION FOR THE INPUT AND OUTPUT FILES OF THE PROGRAM. IF NO LABEL EQUATION FOR THOSE FILES IS USED THEN BY DEFAULT LABEL EQUATION IS AS FOLLOWS:

NEWDISK	IS LABEL EQUATED TO THE NEWLY CREATED FILE PATCHES/ <PROGRAM NAME>
OLDDISK	IS LABEL EQUATED TO THE EXISTING FILE PATCHES/ <PROGRAM NAME>
NEWPATCHES	IS LABEL EQUATED TO THE EXISTING FILE PATCH/ <PROGRAM NAME>
CARDPATCHES	IS LABEL EQUATED TO THE CARD OUTPUT FILE PATCH/ <PROGRAM NAME>

CHANGE NO. 3 (14 CARDS).

ERR MSG FOR NON-NUMERIC SEQ #

THIS CHANGE ADDS A NEW ERROR MESSAGE TO THE PROGRAM IF NON-NUMERIC

SEQUENCE NUMBERS ARE ENCOUNTERED, A MESSAGE OF THE FORM
xxxxxxxALPHA NUMERIC SEQUENCE NUMBERS NOT ALLOWEDxxxxxxERROR<NUMBER>
IS PRINTED OUT.

CHANGE NO. 4 (2 CARDS).

ERR MSG WAS OVERPRINTING

THIS CHANGE ELIMINATES AN ERRONEOUS SITUATION IN WHICH AN ERROR MESSAGE HAS OVERWRITTEN THE PRINTER IMAGE OF THE $# CARD.

CHANGE NO. 5 (82 CARDS).

THIS CHANGE CAUSES PATCH/MERGE TO ACCEPT PATCHES FOR A COBOL OR A COBOL68 PROGRAM, I.E. WITH 6-DIGIT SEQUENCE NUMBERS IN COLUMNS 1 THROUGH 6, INSTEAD OF 8 DIGIT SEQUENCE NUMBERS IN COLUMNS 73 THROUGH 80. THE $@ COBOL OPTION CARD MAY BE USED FOR THIS PURPOSE. WITH THIS OPTION SET PATCH/MERGE REQUIRES THAT THE VOID CARD MUST HAVE ITS "$" IN COLUMN 7 AND THE VOIDING SEQUENCE AND RANGE MUST BE SIX DIGITS IN LENGTH.

CHANGE NO. 6 (14 CARDS).

STATUS OF INPUT FILES

THIS CHANGE PRINTS OUT THE STATUS OF INPUT FILES USED BY PATCH/MERGE. IN PARTICULAR, THE PATCH/MERGE USER IS NOTIFIED WHETHER THE FILES PATCHES/<PROGRAM NAME> AND PATCH/<PROGRAM NAME> ARE NOT DISK, OR LOCKED AS A RESULT OF A SECURITY CONDITION OR PRESENT AND BEING MERGED.

CHANGE NO. 7 (5 CARDS).

HEADER AND LEVEL CARDS

THIS CHANGE UPDATES THE HEADER AND LEVEL CARDS.

CHANGE NO. 8 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

SYSDISK CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

AUXDATA CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

MESSGEN CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

ROTO CHANGES 1 AND 2.

CHANGE NO. 1 (1 CARD).

DATE CARD

THIS CHANGE UPDATES THE DATE CARD.

CHANGE NO. 2 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

STATS1 CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

STATS2 CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

STATS3 CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

STATS4 CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

OLMAINT CHANGES 1 THROUGH 3.

CHANGE NO. 1 (1 CARD).

LEVEL CARD

THIS CHANGE UPDATES THE LEVEL CARD.

CHANGE NO. 2 (6 CARDS).

BACK UP UNIT SWITCH

THIS CHANGE CORRECTS AN ERROR IN WHICH THE BACK UP TAPE AND DISK TESTS WERE SWITCHED SO THAT IF TAPE WAS REQUESTED DISK WOULD BE USED AND VICE-VERSA. THIS CHANGE ALSO CHANGES THE CARD FILE DECLARATION TO ALLOW FOR THE READER TO BE LABEL EQUATED TO DISK FOR INPUT (FOR EXAMPLE, A PATCH FILE ON DISK FOR THE SIMPL COMPILER).

CHANGE NO. 3 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

TPECNF CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

LOGAN CHANGES 1 AND 2.

CHANGE NO. 1 (1 CARD).

HEADER CARD

THIS CHANGE UPDATES THE HEADER CARD.

CHANGE NO. 2 (5 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

MLUGAN CHANGES 1 AND 2.

CHANGE NO. 1 (9 CARDS).

ERRONEOUS STARTING LOG DATES

THIS CHANGE CORRECTS AN ERROR IN WHICH ERRONEOUS STARTING DATES ARE PRINTED FOR MAINTENANCE LOGS DUE TO PICKING UP THE STARTING DATE FROM THE WRONG TYPE OF LOG ENTRY. THIS OCCURRED WHEN ONE MAINTENANCE LOG BECAME FULL AND ANOTHER LOG WAS STARTED TO HOLD THE OVERFLOW ENTRIES.

CHANGE NO. 2 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

LOGOUT CHANGES 1 AND 2.

CHANGE NO. 1 (1 CARD).

HEADER CARD

THIS CHANGE UPDATES THE HEADER CARD.

CHANGE NO. 2 (5 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

LOGOUTR CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

DCFILL CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

DUMPANL CHANGES 1 AND 2.

CHANGE NO. 1 (9 CARDS).

MEMORY LINK VERIFICATION

THIS CHANGE IS NECESSITATED BY THE MCP CHANGE WHICH REMOVED THE VARIABLES MSTART AND MEND FROM THE MCP=S PRT. THE MEMORY LINK CHECK ROUTINE IN DUMP/ANALYZE STILL USES REFERENCES TO THESE VARIABLES RESULTING IN ERRONEOUS BAD LINK INDICATIONS. THIS CHANGE ALSO CORRECTS A PROBLEM WITH ANALYZING A JOB WHICH HAS BEEN ST=ED. THE PRTROW DESCRIPTOR WILL NOW HAVE BITS [3:4]=2, WHICH FAILED THE EXISTING DESCRIPTOR TEST.

CHANGE NO. 2 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

TSFILL CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

TSDUMP CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

COOL CHANGES 1 THROUGH 4.

CHANGE NO. 1 (1 CARD).

ZERO FILE ID WHEN REMOVING

THIS CHANGE CAUSES THE FILE ID OF A FILE BEING REMOVED BY COOL/START TO BE SET TO ZERO. PREVIOUSLY, IT WAS LEFT UNCHANGED, WHICH COULD CAUSE THE MCP TO HANG DURING THE HALT/LOAD IF THE FLAG BIT WAS SET IN THAT WORD.

CHANGE NO. 2 (2 CARDS).

COLD START EOF PROBLEM

THIS CHANGE CORRECTS END OF FILE POINTER CALCULATION FOR FILES DECLARED IN COLD START.

CHANGE NO. 3 (42 CARDS).

COOL START GENERAL CLEAN UP

THIS CHANGE PERFORMS THREE GENERAL FUNCTIONS:

1. CLEANS UP CODE IN BOTH COOL AND COLD START
2. PREVIOUSLY, IF AN ERROR EXIT WAS TAKEN FROM THE CODE SCANNING "REMOVEF" NAME PAIRS FROM CARDS, COOL START WOULD BECOME LOST. THIS PATCH CORRECTS THAT PROBLEM.
3. INSTEAD OF CHANGING THE CHECK IN DIRECTORYTOP FOR A FAULTY PBD NUMBER THE CODE HAS BEEN ELIMINATED BECAUSE THE PBD NUMBER IS NOW KEPT IN DIRECTORYTOP + 3.

CHANGE NO. 4 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

KERNEL CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

MEMDUMP CHANGE 1.

CHANGE NO. 1 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

DSKDSK CHANGES 1 AND 2.

CHANGE NO. 1 (3 CARDS).

OPERATION WITHOUT SPO

THIS CHANGE ALLOWS THE PROGRAM TO COMPLETE ITS OPERATION IN THE ABSENCE OF AN ACTIVE SPO.

CHANGE NO. 2 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

TAPEDSK CHANGES 1 AND 2.

CHANGE NO. 1 (4 CARDS).

OPERATION WITHOUT SPO

THIS CHANGE ALLOWS THE PROGRAM TO COMPLETE ITS OPERATION IN THE ABSENCE OF AN ACTIVE SPO.

CHANGE NO. 2 (4 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

AUXTST CHANGE 1.

CHANGE NO. 1 (5 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

CHECKAL CHANGE 1.

CHANGE NO. 1 (5 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

MASTEST CHANGES 1 AND 2.

CHANGE NO. 1 (2 CARDS).

HEADER CARD

THIS CHANGE UPDATES THE HEADER CARD.

CHANGE NO. 2 (6 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

MAKCAST CHANGE 1.

CHANGE NO. 1 (5 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

AFILTER CHANGE 1.

CHANGE NO. 1 (5 CARDS).

COPY RIGHT NOTICE

THIS CHANGE UPDATES THE COPY RIGHT NOTICE.

TEMPORARY CHANGES TO MARK XVI.0 SYSTEM

MCP CHANGES.

CHANGE NO. 101 (011 CARDS).

THIS PATCH CORRECTS A COBOL68 PROBLEM WITH OPENING FILES IN AN INPUT OR OUTPUT PROCEDURE OF A SORT.

CHANGE NO. 102 (1 CARD).

THIS PATCH PREVENTS PROBLEMS CAUSED BY OPENING A DISK FILE WHOSE BUFFER EXCEEDS 1890 WORDS.

CHANGE NO. 103 (2 CARDS).

THIS PATCH CORRECTS THE HANDLING OF "SM" AND "HM" INPUT MESSAGES.

CHANGE NO. 104 (001 CARD).

THIS PATCH REFORMATS THE "SYSTEM HANG" MESSAGE.

CHANGE NO. 105 (1 CARD).

THIS PATCH CORRECTS AN ERROR IN THE MCP PROCEDURE BACKCLOSE THAT WOULD SOMETIMES CAUSE "PUD" FILES NOT TO BE PUNCHED EVEN WITH THE OPTION AUTOPRNT SET.

CHANGE NO. 106 (2 CARDS).

THIS PATCH CORRECTS SEVERAL PROBLEMS WITH THE BREAKOUT/RESTART MCP OPTION.

CHANGE NO. 107 (1 CARD).

THIS PATCH CORRECTS AN ERROR THAT WOULD CAUSE AN INVALID INDEX IN

CONTROL STATE.

CHANGE NO. 108 (1 CARD).

THIS PATCH WILL ALLOW FOR BETTER SCANNING OF PB KEYIN. IT IS NOW OK TO HAVE SPACES AFTER "#" OR "=" IN THE PB KEYIN REQUEST.

CHANGE NO. 109 (2 CARDS).

THIS PATCH WILL ADD THE I/O CHANNEL NUMBER TO THE ROTO OUTPUT

CHANGE NO. 110 (90 CARDS).

THIS PATCH CORRECTS SEVERAL PROBLEMS IN COBOL68.I.P.C.

1) USING A PACKETS MCP = THE TASKS WOULD NOT BE KEPT WITHIN THE PACKET.

2) IF MORE THAN ONE TYPE=2 PASS=BY=VALUE SEGMENTS WERE PASSED TO A TASK THEN ONLY THE LAST ONE WOULD BE RECEIVED.

3) THE CALLED TASK WOULD HAVE A USERCODE OF ALL ZEROS.

4) AN I.P.C CODE FILE COULD BE EXECUTED EVEN IF IT SHOULD BE RECEIVING PARAMETERS. NOW IT IS FLAGGED AS NOT EXECUTABLE CODE.

5) THE BEGINING AND ENDING OF JOB FOR TASKS ARE NOW MARKED AS BOT AND EOT.

6) CORRECTS A BAD CALL ON MAKEPRESENT THAT WOULD CAUSE A SYSTEM HANG WHEN THE PASS=BY=VALUE TYPE=2 SEGMENTS WERE NOT PRESENT IN CORE. THIS WOULD ONLY SHOW UP IF THE SYSTEM WERE RUNNING ANOTHER JOB AND THAT JOB CAUSED YOUR ARRAY TO BE OVERLAYED.

7) THE COMMON VALUE AND LABEL EQUATION ENTRIES PRESENTED WHEN THE PRUGRAM WAS COMPILED WERE NOT ENTERED INTO THE SHEET, THEREFORE THEY WERE NOT PROCESSED BY THE SELECTION ROUTINES.

8) THERE WAS NO CONTROL CARD IN THE PACKET PAGE FOR THE INVOKED

TASK. NOW ONE IS BUILT AND PLACED THERE.

9) CORRECTS ANOTHER BAD CALL ON MAKEPRESENT THAT WOULD ALSO CAUSE A SYSTEM HANG. THIS TIME IF THE AIT ARRAY WERE NOT PRESENT.

10) CORRECTS A PROBLEM IN ADDRESSING THE TASK ARRAY DESC. "(*P (TSX INX IRP))" IS NOT THE SAME AS "(*P([TRP [TSX]]))".

11) ALLOWS BOTH AN INVOKING TASK AND AN INVOKED TASK TO BE DSED FASTER BY REPLACING TWO "SLEEPS" WITH TWO "COMPLEXSLEEPS" ON TERMIX.

12) THE PROCESSOR AND I/O LIMITS ALONG WITH THE STACK SIZE WERE NOT ENTERED INTO THE SHEET.

CHANGE NO. 111 (10 CARDS).

IF A PARTIAL WORD TRANSFER FROM TAPE IS DONE THIS PATCH WILL MARK THE ERROR AS A PARITY CONDITION SO THAT RETRIES WILL OCCUR UNTIL THE FULL WORD IS CORRECTLY TRANSFERED.

CHANGE NO. 112 (1 CARD).

THIS PATCH WILL CORRECT A CONDITION THAT COULD MARK THE PACKET PAGE AS BEING IN USE BY THE WRONG SYSTEM AFTER A HALT/LOAD.

CHANGE NO. 113 (3 CARDS).

THIS PATCH WILL CORRECT AN ERROR THAT CAUSED A SYSTEM HALT WHEN THE CONSOLE KEYBOARD MESSAGE "DS A/B" IS ENTERED AND THERE IS NO JOB A/B.

CHANGE NO. 114 (1 CARD).

THIS PATCH WILL ALLOW A DISK SQUASH TO BE STOPPED ONLY AFTER IT HAS FINISHED MOVING A FILE. IF STOPPED SOONER THEN THE AVAILABLE DISK TABLE COULD BE DAMMAGED.

CHANGE NO. 115 (1 CARD).

THIS PATCH CORRECTS A PROBLEM WITH UNITS GOING NOT READY AND THEN NOT BEING PICKED UP AS BEING READY WHEN THEY GO READY

CHANGE NO. 116 (2 CARDS).

WHEN USING A MULTI-FILE PBT THE SECOND AND FOLLOWING FILES WILL SOMETIMES BE PRINTED SEVERAL TIMES. THIS WAS CAUSED BY A CONFLICT IN THE USE OF A FIELD IN THE FPB OF A PROGRAM. THE SAME FIELD THAT IS USED TO HOLD THE NUMBER OF COPIES (FROM A FILE EQUATE) IS ALSO USED TO HOLD THE PRN OF THE TAPE.

CHANGE NO. 117 (1 CARD).

IF THE REEL IS GIVEN ON A FILE(EQUATION) CONTROL CARD IT WILL NOT BE USED BY THE MCP WHEN IT LOOKS FOR A TAPE FILE AT FILE OPEN TIME. THIS IS BECAUSE THE REEL NUMBER IS MOVED FROM THE FIB. NOW THE REEL NUMBER IS ONLY MOVED FROM THE FIB IF IT IS NONZERO

CHANGE NO. 118 (1 CARD).

IF A PROGRAM READS A DISK FILE THAT HAS A ROW SIZE THAT IS LESS THAN THE PROGRAMS BUFFER SIZE, THE MCP WILL ONLY GET A CORE BUFFER THE SIZE OF THE DISK FILE'S ROW. BUT ANY PROGRAM THAT USES THE BUFFER WITHOUT GOING THROUGH THE INTRINSICS WILL NOT KNOW THAT THE BUFFER SIZE HAS BEEN CHANGED; THEREFORE, AN INVALID LINK MAY OCCURE IF ONE TRIES TO USE THE PART OF THE BUFFER THAT IS NOT THERE.

CHANGE NO. 119 (2 CARDS).

THIS PATCH CORRECTS A PROBLEM THAT WOULD CAUSE SOME FILES TO BE MARKED IN USE AFTER A HALT/LOAD ON A SHAREDISK SYSTEM.

CHANGE NO. 120 (1 CARD).

IF TWO LIBMAIN/DISKS ARE BOTH WAITING FOR THE SAME TAPE TO BECOME

READY, ONE COPY WILL BE ASSIGNED THE CORRECT TAPE AND THE OTHER COPY OF LIBMAIN WILL BE ASSIGNED TO MTA.

CHANGE NO. 121 (34 CARDS).

B6700 TAPES WITH NEW HEADERS THIS CHANGE ALLOWS THE B5700 LIBRARY MAINTENANCE TO LOAD FILES FROM TAPES WHICH HAVE BEEN CREATED ON A B6700 WITH NEW HEADER FORMATS OF TYPE 2 AND 3.

CHANGE NO. 122 (27 CARDS).

THIS PATCH WILL PLACE THE OUTPUT UNIT NAME OF A LIBRARY COPY IN WORD 27 OF THE SHEET SO THAT WHEN A "TS" IS DONE THIS NAME WILL BE LISTED ON THE OUTPUT MESSAGE.

CHANGE NO. 123 (11 CARDS).

LIBRARY MAINT, CORRECTIONS

THIS CHANGE PERFORMS THE FOLLOWING:

A. CORRECTS MISHANDLING OF ESPDISK SEGMENTS DURING ABORT.

B. MAKES HEADERS CREATED FROM COPYING B6700 TAPE FILES "NEW" TYPE HEADERS AND

C. DISALLOWS TRANSFERS FROM B6700 SOURCE TAPES DIRECT TO TAPE BECAUSE OF POSSIBLE BAD SEGMENTS PER HOW INFORMATION IN THE B6700 HEADER. ATTEMPTS AT SUCH TRANSFERS WILL CAUSE ABORTION WITH THE FOLLOWING MESSAGE :
"#B6700 TAPE TO TAPE NOT ALLOWED".

WARNING: IN THE "COPY" CONTROL CARD, THE MAXIMUM NUMBER OF FILES PER OUTPUT UNIT MAY BE CONFUSED WITH THE <MFID> OF A FILE IF THAT <MFID> IS A NUMBER; I.E. THE <MFID> OF THE FIRST FILE FOLLOWING THE WORD "COPY" MAY NOT BE A NUMBER.

CHANGE NO. 124 (8 CARDS).

THIS PATCH WILL ALLOW PACKETS AND DECKS TO BE MIXED. ALSO IF THE SPO OPTION "PKTONLY" IS SET THEN ALL DECKS WILL BE USED AS IF THEY WERE PACKETS. THAT IS THE USE OF PKTONLY IS NOW CHANGED SO THAT IT NO LONGER CONTROLS THE LOADING OF DECKS BUT TO CONTROL THEIR USE.

CHANGE NO. 125 (1 CARD).

THIS PATCH WILL REPLACE A LINE OF CODE THAT WAS DROPPED FROM THE SYMBOL FILE.

CHANGE NO. 126 (5 CARDS).

THIS PATCH WILL MAKE THE DATACOM AND TSS MCP COMPATIABLE IN THE AREA OF DISK FILE NAMING. THAT IS IF THE FILES MFID IS ZERO AND THE PROGRAM"S USERCODE IS NON-ZERO THEN THE FILES FID IS MOVED TO THE MFID AND THE USERCODE IS MOVED TO THE FID.

CHANGE NO. 127 (46 CARDS).

THIS PATCH CORRECTS AN ERROR IN THE NEW (MARK.XV) SELECTRUN PROCEDURES. IF SELECTRUN CANNOT GET MEMORY FOR A JOBS STACK AND PRT THE JOB WILL NOT BE SELECTED FOE EXECUTION (EVEN IF XS=ED) BUT SELECTRUN LEFT SEVERAL MCP TABLES SETUP AS IF IT HAD SELECTED THE JOB. TO AVOID THIS SITUTATION THE CORE FOR THE PROGRAMS STACK AND PRT IS OBTAINED BEFURE THESE TABLES ARE SETUP.

SEVERAL PROBLEMS CAN HAPPEN WITHOUT THIS PATCH, ONE IS A JOB COULD BECOME PART OF A DIFFERENT PACKET THEN THE ONE IT WAS EXECUTED FROM THUS THE ORIGINAL PACKET WOULD BE LEFT IN USE.

CHANGE NO. 128 (1 CARD).

THIS PATCH CORRECTS AN ERROR IN CONTROLCARD THAT COULD CAUSE UNEXPLAINED CONTROL CARD ERRORS AND SYSTEM HALTS. THE ERROR WAS CAUSED BY CONTROLCARD NOT RESETTING THE STACK VARIABLE "PROCVAL" BACK TO ZERO AFTER A TYPED PROCEDURE WAS CALLED. THUS THE NEXT TYPED PROCEDURE CALLED WOULD START WITH A NON ZERO VALUE.

CHANGE NO. 129 (3 CARDS).

THIS PATCH WILL CORRECT THE WAY CONTROL CARD HANDLES PACKET ERRORS. WITHOUT THIS PATCH THE PACKET WILL ONLY BE FLUSHED TO THE NEXT END OR WAIT CARD NOT TO JUST THE END CARD.

CHANGE NO. 130 (3 CARDS).

THIS PATCH WILL CORRECT AN ERROR IN THE NEW LIBMAIN/DISK PROGRAM. IF THERE WAS A "NULL LIBRARY TRANSFER" THEN IF THE INPUT OR OUTPUT UNIT IS DISK, WORD ONE OF MEMORY WOULD BE OVERWRITTEN BY A ZERO AS THE MCP TRIED TO CLOSE THE UNIT. ONE OF THE MANY PROBLEMS CAUSED BY THIS SITUTATION WAS THAT NO USER CODES WOULD BE ACCEPTED BY CONTROLCARD BECAUSE THE MCP"S USERCODE(STORED IN WORD ONE) IS ZERO. NO ERROR MESSAGE WAS GIVEN TO INDICATE THIS CONDITION AND THE SYSTEM CONTINUED TO RUN BUT ALL USERCODES WOULD BE ZERO.

CHANGE NO. 131 (6 CARDS).

THIS PATCH WILL CAUSE PRNPBT/DISK AND AUTO-LDCNTRL TO BE EXECUTED WITH THE MCP"S USERCODE. THIS INFORMATION IS PLACED INTO THE LOG FILE. ALSO WITH THIS PATCH PACKET PAGES WILL BE GIVEN THE MCP"S USER CODE NOT THE USER CODE "PACKET".

CHANGE NO. 132 (2 CARDS).

THIS PATCH WILL ADD THE USERCODE OF THE PROGRAM THAT CREATED A PRINTER BACKUP FILE(PBT OR PBD) TO THE LOG. WITH OUT THIS PATCH A LOG PROGRAM IS NOT ABLE TO TELL WHO THE CREATOR OF THE FILE IS. WORD 15 OF THE CONTROL RECORD IS USED TO HOLD THE USERCODE. THIS WORD WAS UNUSED BEFORE.

CHANGE NO. 133 (2 CARDS).

THIS PATCH WILL GIVE A CONTROL CARD ERROR IF EATHER THE EQUAL SIGN OR USER CODE IS MISSING ON A USER CONTROL CARD.

CHANGE NO. 134 (1 CARD).

THIS PATCH WILL CORRECT A CONDITION IN LIBMAIN/DISK THAT WOULD CAUSE A LIBRARY TAPE TO BE CREATED WITH NO LIBRARY FILES ON IT. THIS TAPE RESULTED ON A NULL LIBRARY TRANSFER. ALSO CORE SPACE FOR THE BUFFERS IS NOT OBTAINED UNLESS THERE ARE SOME FILES TO BE COPIED.

CHANGE NO. 135 (1 CARD).

WITH SOME SYSTEMS THE FIRST I/O DONE TO A PRINTER FOR THE PACKET PAGE WILL CAUSE THE PRINTER TO HANG UP FOR EXTENDED PERIOD OF TIME. THIS WAS CAUSED BY DOING AN INVALID PRINTER OPERATION. THAT IS NO PRINTING AND NO PAPER MOVEMENT.

THIS I/O RESULTED WHEN THE MCP CHANGED THE I/O DESCRIPTOR FOR THE ABORTED LINE SO THAT THIS LINE WOULD NOT BE PRINTED.

NOW A SINGLE SPACE IS GENERATED.

THE MCP WOULD RECOVER FROM THIS CONDITION AFTER ABOUT 20 SECONDS.

CHANGE NO. 136 (3 CARDS).

THIS PATCH CORRECTS AN ERROR IN PATCH NUMBER 127 THAT WOULD NOT ALLOW RESTART JOBS TO BE RESTARTED.

CHANGE NO. 137 (1 CARD).

THIS PATCH CAUSES HEADER SPACE FOR DUPLICATE DISK FILES TO BE FORGOTTEN. PREVIOUSLY, IF THE LIBRARY MAINTENANCE TASK INVOLVED MANY DUPLICATIONS, THE HEADER SPACE OF EACH DUPLICATE FILE WAS LEFT IN CORE EVENTUALLY RESULTING IN A "NO-MEM" CONDITION ON MIX ZERO.

CHANGE NO. 138 (5 CARDS).

THIS PATCH CORRECTS SEVERAL ERRORS IN THE MCP COMPILE TIME OPTION "WORKSET". THEY ARE.

1) AN AUTO-OK FOR THE WRONG JOB IS SOMETIMES DONE. THIS COULD HAPPEN IF A STOPPED JOB WAS OK-ED BY THE OPERATOR

2) UNDER SOME CONDITIONS SELECTION COULD PLACE A JOB IN THE MIX EVEN IF A JOB HAD BEEN AUTO STOPPED.

3) THE FIRST JOB TO BE AUTO-OK-ED WAS NOT THE LAST JOB AUTO-STOPPED AS IT SHOULD HAVE BEEN.

CHANGE NO. 139 (48 CARDS).

THIS PATCH ADDS THE ABILITY TO FORCE A BREAKOUT FROM THE SPO. THE OPERATOR CAN NOW ENTER "EI" TO CAUSE BREAK FILES TO BE BUILT FOR ALL JOBS IN THE MIX. OR <MIX>EI IF ONLY A BREAK FILE FOR ONE JOB IS NEEDED.

NOTE: MCP JOBS(PRNPBT, LOCNTRL, LIBMAIN) AND COMPILERS CANNOT BE BORKEN. AND IF THE "EI" FORM IS USED THEN ALL JOBS THAT ARE BROKEN WILL ALSO BE DS-ED AND THE CORE FACTOR IS SET TO ZERO SO THAT NO OTHER JOBS WILL ENTER THE MIX.

CHANGE NO. 140 (8 CARDS).

THIS PATCH CORRECTS AN ERROR IN THE NEW SELECTION PROCEDURE THAT WOULD ALLOW A JOB WITH A LOWER PRIORITY TO BE SELECTED FOR EXECUTION IF A HIGHER PRIORITY JOB WAS WAITIING FOR CORE. ALSO IF THE MCP COMPILE TIME OPTION "BREAKOUT" WAS INCLUDED IN THE MCP THEN SELECTION WOULD CORRECTLY PROCESS THE JOB BUT A MISLEADING "REASON" FOR A JOB BEING SCHEDULED WOULD SOMETIMES BE GIVEN(THE REASON WAS "RESTART IN PROGRESS") AND WOULD BE GIVEN WHEN A HIGHER PRIORITY JOB WAS SCHEDULED AND A LOWER PRIORITY JOB WAS SCHEDULED BECAUES OF IT.

CHANGE NO. 141 (3 CARDS).

THIS PATCH WILL INSURE THAT USERSTA IS ZEROED OUT SO THAT A PROGRAM DOSN"T ACCIDENTALLY GET ATTACHED TO A TERMINAL.

CHANGE NO. 142 (32 CARDS).

THIS PATCH WILL ALLOW WORKSET TO BETTER CONTROL THE SYSTEM BY GIVEING IT THE ABILITY TO START JOBS IF THE OLAY RATE DROPS OFF. NOW JOBS MAY BE AUTO OK-ED EVEN IF A DIFFERENT JOB HAS NOT LEFT THE MIX AS WAS BEFORE REQUIRED.

CHANGE NO. 143 (15 CARDS).

THIS PATCH WILL SAVE THE WORKSET PARAMETERS ON DISK AND RESET THEM

ON A HALT/LOAD, THE VALUE OF "WKSETCYCLETIME","WKSETINSTRUCT", "WKSETOLERANCE","WKSETMAXOLAY" ARE STORED IN THE DIRECTORYTOP+3 FOR EACH SYSTEM.

CHANGE NO. 144 (1 CARD).

THIS PATCH CORRECTS A CONDITION THAT WOULD CAUSE AN INVALID ADDRESS ON A HALT LOAD IF MEMORY MOD ONE IS OFF LINE.

CHANGE NO. 145 (25 CARDS).

THIS PATCH ADDS THREE NEW TIME FUNCTIONS:

1) TIME(-3), WILL RETURN THE CURRENT STATUS OF THE PACKETERR BIT

2) TIME(-4), THIS FUNCTION WILL SET THE PACKETERR BIT

3) TIME(-5), WILL RETURN THE CURRENT VALUE OF PACKETACT.

THESE THREE FUNCTIONS WERE ADDED TO GIVE A PROGRAM SOME CONTROL OVER THE PACKET. NOW A PROGRAM CAN "KILL" A PACKET BY A TIME FUNCTION RATHER THAN BY DS-ING ITSELF WITH A RUN TIME ERROR(DIV BY ZERO, ETC). ALSO A PROGRAM CAN "SEE" IF A SISTER PROGRAM HAS RUN INTO TROUBLE, AND A PROGRAM CAN TELL HOW MANY JOBS ARE RUNNING FROM THE PACKET AT THIS TIME.

TIME(-4) WILL RETURN THE VALUE OF PACKETERR BEFORE IT SETS IT.

EXAMPLES: (ALGOL)

```
IF A LSS 0 THEN          % WE HAVE A PROBLEM
   BEGIN                 % LETS KILL THIS RUN
   B:=TIME(-4);          % KILL PACKET
   GO TO EXIT;           % EXIT PROGRAM
   END;                  %
```

AND IN A PROGRAM RUN FROM THE SAME PACKET AT THE SAME TIME

```
   IF BOOLEAN(TIME(-3)) THEN GO TO EXIT; % ERROR IN OTHER PGM.
```

IF THE FIRST PROGRAM FINDS A PROBLEM AND WANTS TO STOP THIS RUN IT SETS THE PACKETERR BIT THEN THE SECOND PROGRAM WILL "SEE" THIS AND ALSO GO TO EOJ.

CHANGE NO. 146 (14 CARDS).

THIS PATCH WILL ALLOW UNITS TO BE SAVED ACROSS A HALT/LOAD. THE VALUE OF SAVEWORD IS NOW KEPT IN THE DIRECTORYTOP SEGMENT FOR EACH SYSTEM(WORD 29 IS USED). UNITS MARKED "TO BE SAVED" WILL BE BUT UNITS THAT ARE "RW/L" WILL NOT BE SAVED. ONLY THE UNITS "SV-ED" BY THE OPERATOR ARE ACTUALLY SAVED ACROSS THE HALT/LOAD.

CHANGE NO. 147 (3 CARDS).

THIS PATCH WILL SET THE PACKETERR BIT IF THERE IS A ZIP ERROR IN THE PACKET.

CHANGE NO. 148 (82 CARDS).

THIS PATCH WILL ALLOW THE SYSTEM OPERATOR TO COPY TO OR FROM A LIBRARY TAPE BY UNIT NAME. THE COPY CONTAOL CARD NOW CONTAINS AN OPTIONAL "ON" CLASE.

EXAMPLE:

<I> COPY MY/FILE FROM TAPE1 ON MTB.
<I> COPY DISK/FILE FROM DISK TO SAVEIT ON MTD.

IF THE NAMED UNIT IS NOT AVAILABLE THEN A MESSAGE WILL BE GIVEN AT THE SPO AND THE OPERATOR MY EATHER PROVIDE THE UNIT OR TAKE WHATEVER ACTION IS NEEDED. THE MESSAGE IS "NO FIL ON MTN" OR "MT ROD ON MTN" WHERE MTN IS THE NAMED UNIT.

CHANGE NO. 149 (25 CARDS).

THIS PATCH ADDS A NEW SPO OPTION "LIBERR". THIS OPTION CONTROLS THE PRINTING ON LIBRARY ERROR MESSAGES. THAT IS THE NOT REMOVED MESSAGES THAT ARE GENERATED FROM A PACKET IF IT CONTAINS A REMOVE CARD AND THE FILE IS NOT ON DISK. THE OPERATOR WILL ONLY GET THESE MESSAGES IF THE OPTION LIBERR IS SET BUT THE ERROR MESSAGE ALWAYS GOES INTO THE PACKET OUTPUT.

THIS OPTION WAS ADDED TO KEEP THE SPO FREE FOR MORE USEFUL WORK.

CHANGE NO. 150 (22 CARDS).

THIS PATCH CHANGES THE USE OF THE SPO OPTION "SEPARATE". IT NOW IS USED TO CONTROL THE PRINTER SPACEING AROUND THE LABLE PAGE. THAT IS IF SEPARATE IS RESET THEN ONLY A DOUBLE SPACE IS DONE AFTER THE BEGINING LABEL AND BEFORE THE ENDING LABEL. IF IT IS SET THE OPERATION OF THE PRINTER IS AS IT WAS BEFORE(A FULL PAGE FOR THE LABEL RECORD).

THIS WAS DONE TO HELP WITH THE ENERGY CRUNCH AND PER CUBE REQUEST.

CHANGE NO. 201 (1 CARD).

THIS PATCH WILL ENABLE THE USE OF AN "SM" KEYBOARD INPUT MESSAGE AFTER THE USE OF AN "HM" KEYBOARD INPUT MESSAGE.

CHANGE NO. 202 (1 CARD).

THIS PATCH KEEPS JOBS WHOSE PRT EXCEEDS 1023 WORDS FROM BEING EXECUTED.

CHANGE NO. 203 (4 CARDS).

THIS PATCH WILL CORRECT A CONDITION THAT CAUSED MISHANDLING OF SOME DATACOM FUNCTIONS.

CHANGE NO. 204 (25 CARDS).

THIS PATCH SHOULD BE IMPLEMENTED WHEN OVERWRITES OF DISK ADDRESS ZERO ARE OCCURRING. THIS IS A DIAGNOSTIC PATCH AND WILL HALT THE SYSTEM BEFORE THE OVERWRITE OCCURS. NO ERROR MESSAGE WILL BE GIVEN WHEN THE SYSTEM HANGS.

CHANGE NO. 206 (1 CARD).

THIS PATCH WILL ELIMINATE AN EXTRANOUS CARD FROM THE OUTPUT DECK OF AN UNLABELED PUNCH FILE.

CHANGE NO. 207 (2 CARDS).

THIS PATCH CORRECTS THE PROBLEM OF SYSTEM FILE DISK DIRECTORY HEADERS BEING MARKED IN-USE AFTER A PROGRAM HAD PERFORMED A SEARCH AGAINST THEM.

CHANGE NO. 301 (2 CARDS).

THIS PATCH ALLOWS READING OF PURE BINARY CARD INPUT. IF THE FILE IS DECLARED AS ALPHA WITH A BUFFER LENGTH OF 20 WORDS, ALSO THE CARD READER WILL BE MARKED SAVED WHEN THE PROGRAM CLOSES THE FILE. CARE SHOULD BE TAKEN WHEN USING THIS FEATURE SINCE A "END" CARD WILL NOT BE SEEN BY THE SYSTEM. IT IS THE PROGRAMS RESPONSIBILITY TO DETECT THAT THE END OF FILE HAS OCCURED.

TSSMCP CHANGES.

CHANGE NO. 101 (001 CARD).

THIS PATCH REFORMATS THE "SYSTEM HANG" MESSAGE.

CHANGE NO. 102 (010 CARDS).

THIS CHANGE CORRECTS AN ERROR IN THE HANDLING OF COBOL68 OPEN STATEMENTS THAT OCCUR IN THE INPUT OR OUTPUT PROCEDURE OF A SORT. THE FILES WILL NOW BE LEFT OPEN AS THEY SHOULD.

CHANGE NO. 103 (1 CARD).

THIS PATCH CORRECTS AN ERROR IN THE MCP PROCEDURE BACKCLOSE THAT WOULD SOMETIMES CAUSE "PUD" FILES NOT TO BE PUNCHED EVEN WITH THE OPTION AUTOPRNT SET.

CHANGE NO. 104 (1 CARD).

FIXES THE SCHEDULE LINE "XS" KEYIN

CHANGE NO. 105 (2 CARDS).

THIS PATCH CORRECTS AN INVALID INDEX IN CONTROL STATE.

CHANGE NO. 106 (8 CARDS).

THIS PATCH WILL ALLOW PACKETS TO BE MIXED. ALSO IF THE SPO OPTION "PKTONLY" IS SET THEN ALL DECKS WILL BE USED AS IF THEY WERE PACKETS. THAT IS THE USE OF PKTONLY IS NOW CHANGED SO THAT IT NO LONGER CONTROLS THE LOADING OF DECKS BUT TO CONTROL THEIR USE.

CHANGE NO. 107 (1 CARD).

THIS PATCH WILL CORRECT A CONDITION THAT COULD MARK THE PACKET PAGE AS BEING IN USE BY THE WRONG SYSTEM AFTER A HALT/LOAD.

CHANGE NO. 108 (001 CARD).

THIS PATCH ALLOWS ERROR MESSAGES LONGER THAN 60 CHARACTERS TO BE PRINTED CORRECTLY ON A REMOTE TERMINAL THROUGH CANDE.

CHANGE NO. 109 (1 CARD).

THIS PATCH REPLACES A LINE OF CODE THAT WAS INADVERTENTLY LEFT OUT IN THE XV.3 RELEASE. WITHOUT THIS PATCH, ESPDISK IS NOT HANDLED CORRECTLY, INCORRECT LOG ENTRIES ARE MADE, AND SYSTEM HALTS CAN OCCUR.

CHANGE NO. 110 (1 CARD).

THIS PATCH CORRECTS THE TSSMCP"S HANDLING OF TAPE FILES CLOSED WITH THE "CLOSE(<FILEID>,*)" SYNTAX. THE TSSMCP WOULD "FORGET" THAT THE LAST I/O DONE ON THIS FILE WAS A REVERSE I/O BY ZEROING FIB[16], THUS WHEN THE FILE IS OPENED REVERSE AGAIN THE TSSMCP WOULD SPACE (FORWARD) OVER THE NEXT FILE, AND THE FILE THAT WAS JUST CLOSED WOULD BE ACCESSED AGAIN.

CHANGE NO. 111 (1 CARD).

THIS PATCH WILL ALLOW A DISK SQUASH TO BE STOPPED ONLY AFTER IT HAS FINISHED MOVING A FILE. IF STOPPED SOONER THEN THE AVAILABLE DISK TABLE COULD BE DAMMAGED.

CHANGE NO. 112 (1 CARD).

THIS PATCH CORRECTS A PROBLEM WITH UNITS GOING NOT READY AND THEN NOT BEING PICKED UP AS BEING READY WHEN THEY GO READY

CHANGE NO. 113 (5 CARDS).

THIS PATCH CORRECTS A CONDITION THAT CAUSED THE LOG OFF MESSAGE TO CONTAIN ERRONEOUS DATA. AS A RESULT OF THE NEW MCP COMPILE TIME OPTION "NEWLOGGING" THE VALUE OF PROCTIME[P1MIX] IS CHANGED TO INDICATE IF THE PROGRAM SHOULD BE CHARGED FOR CPU TIME OR NOT. THIS WAS NOT TAKEN INTO CONSIDERATION WHEN COMM2 PASSED THE VALUE TO BE USED AS "CLOCK" BY CANDE.

CHANGE NO. 114 (2 CARDS).

WHEN USING A MULTI-FILE PBT THE SECOND AND FOLLOWING FILES WILL SOMETIMES BE PRINTED SEVERAL TIMES. THIS WAS CAUSED BY A CONFLICT IN THE USE OF A FIELD IN THE FPB OF A PROGRAM. THE SAME FIELD THAT IS USED TO HOLD THE NUMBER OF COPIES (FROM A FILE EQUATE) IS ALSO USED TO HOLD THE PRN OF THE TAPE. THIS PROBLEM IS FIXED BY NOT MOVING THE PRN TO THE FBP UNTIL THE FILE IS CLOSED. THEN WHEN THE FILE IS REOPENED A NEW SECTION OF THE FPB WILL BE USED AND THIS WILL CONTAIN THE CORRECT VALUE FOR COPIES.

CHANGE NO. 115 (1 CARD).

IF THE REEL IS GIVEN ON A FILE(EQUATION) CONTROL CARD IT WILL NOT BE USED BY THE MCP WHEN IT LOOKS FOR A TAPE FILE AT FILE OPEN TIME. THIS IS BECAUSE THE REEL NUMBER IS MOVED FROM THE FIB. NOW THE REEL NUMBER IS ONLY MOVED FROM THE FIB IF IT IS NONZERO

CHANGE NO. 116 (1 CARD).

IF A PROGRAM READS A DISK FILE THAT HAS A ROW SIZE THAT IS LESS THAN THE PROGRAMS BUFFER SIZE, THE MCP WILL ONLY GET A CORE BUFFER THE SIZE OF THE DISK FILE"S ROW. BUT ANY PROGRAM THAT USES THE BUFFER WITHOUT GOING THROUGH THE INTRINSICS WILL NOT KNOW THAT THE BUFFER SIZE HAS BEEN CHANGED; THEREFORE, AN INVALID LINK MAY OCCURE IF ONE TRIES TO USE THE PART OF THE BUFFER THAT IS NOT THERE.

CHANGE NO. 117 (2 CARDS).

THIS PATCH CORRECTS A PROBLEM THAT WOULD CAUSE SOME FILES TO BE MARKED IN USE AFTER A HALT/LOAD ON A SHAREDISK SYSTEM.

CHANGE NO. 118 (1 CARD).

IF TWO LIBMAIN/DISKS ARE BOTH WAITING FOR THE SAME TAPE TO BECOME READY, ONE COPY WILL BE ASSIGNED THE CORRECT TAPE AND THE OTHER COPY OF LIBMAIN WILL BE ASSIGNED TO MTA.

CHANGE NO. 119 (34 CARDS).

B6700 TAPES WITH NEW HEADERS THIS CHANGE ALLOWS THE B5700 LIBRARY MAINTENANCE TO LOAD FILES FROM TAPES WHICH HAVE BEEN CREATED ON A B6700 WITH NEW HEADER FORMATS OF TYPE 2 AND 3.

CHANGE NO. 121 (11 CARDS).

LIBRARY MAINT. CORRECTIONS

THIS CHANGE PERFORMS THE FOLLOWING:

A. CORRECTS MISHANDLING OF ESPDISK SEGMENTS DURING ABORT,

B. MAKES HEADERS CREATED FROM COPYING B6700 TAPE FILES "NEW" TYPE HEADERS AND

C. DISALLOWS TRANSFERS FROM B6700 SOURCE TAPES DIRECT TO TAPE BECAUSE OF POSSIBLE BAD SEGMENTS PER ROW INFORMATION IN THE B6700 HEADER, ATTEMPTS AT SUCH TRANSFERS WILL CAUSE ABORTION WITH THE FOLLOWING MESSAGE :
"#B6700 TAPE TO TAPE NOT ALLOWED".

WARNING: IN THE "COPY" CONTROL CARD, THE MAXIMUM NUMBER OF FILES PER OUTPUT UNIT MAY BE CONFUSED WITH THE <MFID> OF A FILE IF THAT <MFID> IS A NUMBER; I.E. THE <MFID> OF THE FIRST FILE FOLLOWING THE WORD "COPY" MAY NOT BE A NUMBER.

CHANGE NO. 122 (1 CARD).

THIS PATCH WILL ALLOW FOR BETTER SCANNING OF PB KEYIN. IT IS NOW OK TO HAVE SPACES AFTER A "#" OR "=" IN PB KEYIN REQUEST.

CHANGE NO. 123 (27 CARDS).

THIS PATCH WILL PLACE THE OUTPUT UNIT NAME OF A LIBRARY COPY IN WORD 27 OF THE SHEET SO THAT WHEN A "TS" IS DONE THIS NAME WILL BE LISTED

ON THE OUTPUT MESSAGE.

CHANGE NO. 124 (2 CARDS).

WHEN USING A MULTI-FILE PBT THE SECOND AND FOLLOWING FILES WILL SOMETIMES BE PRINTED SEVERAL TIMES. THIS WAS CAUSED BY A CONFLICT IN THE USE OF A FIELD IN THE FPB OF A PROGRAM. THE SAME FIELD THAT IS USED TO HOLD THE NUMBER OF COPIES (FROM A FILE EQUATE) IS ALSO USED TO HOLD THE PRN OF THE TAPE.

CHANGE NO. 126 (1 CARD).

THIS PATCH CORRECTS AN ERROR IN CONTROLCARD THAT COULD CAUSE UNEXPLAINED CONTROL CARD ERRORS AND SYSTEM HALTS. THE ERROR WAS CAUSED BY CONTROLCARD NOT RESETTING THE STACK VARIABLE "PROCVAL" BACK TO ZERO AFTER A TYPED PROCEDURE WAS CALLED. THUS THE NEXT TYPED PROCEDURE CALLED WOULD START WITH A NON ZERO VALUE.

CHANGE NO. 127 (3 CARDS).

THIS PATCH WILL CORRECT THE WAY CONTROL CARD HANDLES PACKET ERRORS. WITHOUT THIS PATCH THE PACKET WILL ONLY BE FLUSHED TO THE NEXT END OR WAIT CARD NOT TO JUST THE END CARD.

CHANGE NO. 128 (3 CARDS).

THIS PATCH WILL CORRECT AN ERROR IN THE NEW LIBMAIN/DISK PROGRAM. IF THERE WAS A "NULL LIBRARY TRANSFER" THEN IF THE INPUT OR OUTPUT UNIT IS DISK, WORD ONE OF MEMORY WOULD BE OVERWRITTEN BY A ZERO AS THE MCP TRIED TO CLOSE THE UNIT. ONE OF THE MANY PROBLEMS CAUSED BY THIS SITUTATION WAS THAT NO USER CODES WOULD BE ACCEPTED BY CONTROLCARD BECAUSE THE MCP"S USERCODE(STORED IN WORD ONE) IS ZERO. NO ERROR MESSAGE WAS GIVEN TO INDICATE THIS CONDITION AND THE SYSTEM CONTINUED TO RUN BUT ALL USERCODES WOULD BE ZERO.

CHANGE NO. 129 (6 CARDS).

THIS PATCH WILL CAUSE PRNPBT/DISK AND AUTO-LOCNTRL TO BE EXECUTED

WITH THE MCP"S USERCODE. THIS INFORMATION IS PLACED INTO THE LOG FILE. ALSO WITH THIS PATCH PACKET PAGES WILL BE GIVEN THE MCP"S USER CODE NOT THE USER CODE "PACKET".

CHANGE NO. 131 (2 CARDS).

THIS PATCH WILL GIVE A CONTROL CARD ERROR IF EATHER THE EQUAL SIGN OR USER CODE IS MISSING ON A USER CONTROL CARD.

CHANGE NO. 132 (1 CARD).

THIS PATCH WILL CORRECT A CONDITION IN LIBMAIN/DISK THAT WOULD CAUSE A LIBRARY TAPE TO BE CREATED WITH NO LIBRARY FILES ON IT. THIS TAPE RESULTED ON A NULL LIBRARY TRANSFER. ALSO CORE SPACE FOR THE BUFFERS IS NOT OBTAINED UNLESS THERE ARE SOME FILES TO BE COPIED.

CHANGE NO. 133 (1 CARD).

WITH SOME SYSTEMS THE FIRST I/O DONE TO A PRINTER FOR THE PACKET PAGE WILL CAUSE THE PRINTER TO HANG UP FOR EXTENDED PERIOD OF TIME. THIS WAS CAUSED BY DOING AN INVALID PRINTER OPERATION. THAT IS NO PRINTING AND NO PAPER MOVEMENT.

THIS I/O RESULTED WHEN THE MCP CHANGED THE I/O DESCRIPTOR FOR THE ABORTED LINE SO THAT THIS LINE WOULD NOT BE PRINTED.

NOW A SINGLE SPACE IS GENERATED. THE MCP WOULD RECOVER FROM THIS CONDITION AFTER ABOUT 20 SECONDS.

CHANGE NO. 134 (1 CARD).

THIS PATCH CAUSES HEADER SPACE FOR DUPLICATE DISK FILES TO BE FORGOTTEN. PREVIOUSLY, IF THE LIBRARY MAINTENANCE TASK INVOLVED MANY DUPLICATIONS, THE HEADER SPACE OF EACH DUPLICATE FILE WAS LEFT IN CORE EVENTUALLY RESULTING IN A "NO-MEM" CONDITION ON MIX ZERO.

CHANGE NO. 135 (1 CARD).

THIS PATCH CORRECTS A CONDITION THAT WOULD CAUSE AN INVALID ADDRESS ON A HALT LOAD IF MEMORY MOD ONE IS OFF LINE.

CHANGE NO. 136 (25 CARDS).

THIS PATCH ADDS THREE NEW TIME FUNCTIONS:

1) TIME(-3), WILL RETURN THE CURRENT STATUS OF THE PACKETERR BIT

2) TIME(-4), THIS FUNCTION WILL SET THE PACKETERR BIT

3) TIME(-5), WILL RETURN THE CURRENT VALUE OF PACKETACT.

THESE THREE FUNCTIONS WERE ADDED TO GIVE A PROGRAM SOME CONTROL OVER THE PACKET. NOW A PROGRAM CAN "KILL" A PACKET BY A TIME FUNCTION RATHER THAN BY DS-ING ITSELF WITH A RUN TIME ERROR(DIV BY ZERO, ETC). ALSO A PROGRAM CAN "SEE" IF A SISTER PROGRAM HAS RUN INTO TROUBLE. AND A PROGRAM CAN TELL HOW MANY JOBS ARE RUNNING FROM THE PACKET AT THIS TIME.

TIME(-4) WILL RETURN THE VALUE OF PACKETERR BEFORE IT SETS IT.

EXAMPLES: (ALGOL)

```
IF A LSS 0 THEN          % WE HAVE A PROBLEM
   BEGIN                 % LETS KILL THIS RUN
   B:=TIME(-4);          % KILL PACKET
   GO TO EXIT;           % EXIT PROGRAM
   END;                  %
```

AND IN A PROGRAM RUN FROM THE SAME PACKET AT THE SAME TIME

```
IF BOOLEAN(TIME(-3)) THEN GO TO EXIT; % ERROR IN OTHER PGM.
```

IF THE FIRST PROGRAM FINDS A PROBLEM AND WANTS TO STOP THIS RUN IT SETS THE PACKETERR BIT THEN THE SECOND PROGRAM WILL "SEE" THIS AND ALSO GO TO EOJ.

CHANGE NO. 138 (3 CARDS).

THIS PATCH WILL SET THE PACKETERR BIT IF THERE IS A ZIP ERROR IN THE PACKET.

CHANGE NO. 140 (22 CARDS).

THIS PATCH CHANGES THE USE OF THE SPO OPTION "SEPARATE", IT NOW IS USED TO CONTROL THE PRINTER SPACEING AROUND THE LABLE PAGE, THAT IS IF SEPARATE IS RESET THEN ONLY A DOUBLE SPACE IS DONE AFTER THE BEGINING LABEL AND BEFORE THE ENDING LABEL. IF IT IS SET THE OPERATION OF THE PRINTER IS AS IT WAS BEFORE(A FULL PAGE FOR THE LABEL RECORD).

THIS WAS DONE TO HELP WITH THE ENERGY CRUNCH AND PER CUBE REQUEST.

CHANGE NO. 201 (005 CARDS).

THIS CHANGE CAUSES THE RESULT DESCRIPTOR OF EACH I/O TO BE STORED INTO M[@153+I/O CHANNEL NUMBER].

CHANGE NO. 202 (25 CARDS).

THIS PATCH ALLOWS CELL ZERO TO BE MONITORED FOR INVALID WRITES (AFTER INITIALIZE). IN ORDER TO ENABLE THIS, A NEW TOGLE IS USED , MEMTOG, RATHER THAN M[0].[17:1], TO INTERLOCK MEMORY BELOW THE FENCE. WITH THIS PATCH REFERENCE TO STOREDY MUST BE MADE AS"STOREDY(MIX, TOG);", WHERE MIX IS THE MIX INDEX OF THE PROCESS AND TOG IS EITHER 0 OR 1.

CHANGE NO. 203 (29 CARDS).

THIS CHANGE IS PROVIDED FOR DEBUGGING PURPOSES AND CAUSES THE MCP TO HANG IN A "DO UNTIL FALSE" LOOP WHENEVER DISK SEGMENT ZERO IS ABOUT TO BE UNEXPECTEDLY OVERWRITTEN.

CHANGE NO. 301 (3 CARDS).

THIS PATCH ALLOWS THE READING OF PURE BINARY CARD INPUT. IF THE FILE IS DECLARED AS ALPHA WITH A BUFFER LENGTH OF 20 WORDS. ALSO THE CARD READER WILL BE MARKED SAVED WHEN THE PROGRAM CLOSES THE FILE. CARE SHOULD BE TAKEN WHEN USING THIS FEATURE SINCE A "QEND" CARD WILL NOT BE SEEN BY THE SYSTEM. IT IS THE PROGRAMS RESPONCIBILITY TO DETECT WHEN THE END OF THE FILE HAS OCCURRED.

CHANGE NO. 302 (049 CARDS).

THIS PATCH IMPLEMENTS THE EOF BRANCH IF "QEND" IS TYPED TO A REMOTE JOB AS INPUT. ALSO THE PARITY LABEL WILL BE TAKEN IF DATA IS NOT PRESENT AND A ZERO TIME OUT WAS SPECIFIED.

INTRINSICS CHANGES.

CHANGE NO. 101 (7 CARDS).

THIS PATCH CORRECTS A PROBLEM, WHEREIN, IF THE ALGOL FILE ATTRIBUTE FOR A NOT OPEN, LOCKED FILE WAS INTERROGATED THE VALUE RETURNED WOULD BE ZERO.

CHANGE NO. 102 (005 CARDS).

THIS PATCH FORMATS THE FORTRAN ERROR MESSAGES CORRECTLY.

CHANGE NO. 103 (002 CARDS).

THIS PATCH CORRECTS A CONDITION THAT CAUSED E-TYPE-FORMATS TO OUTPUT "*",S. INSTEAD OF CONVERTING THE FIELD.

CHANGE NO. 104 (001 CARD).

THIS PATCH CORRECTS PROPER FORMATING OF STRING CHARACTERS WHEN USING SEMICOLONS IN BASIC PRINT STATEMENTS.

CHANGE NO. 105 (1 CARD).

THIS PATCH PREVENTS A SYSTEM HANG CAUSED BY THE INTRINSIC COBOLIO TRYING TO DO PROTECT (F.P.M.) I/O TO CARD READER FILES WHEN THE SYSTEM IS OPERATING UNDER LDCNTRL/DISK.

CHANGE NO. 106 (3 CARDS).

THIS PATCH ALLOWS THE RESULT OF A COMPLEX NUMBER RAISED TO A REAL EXPONENT TO HAVE THE CORRECT SIGN FOR NEGATIVE EXPONENTS.

CHANGE NO. 107 (2 CARDS).

THIS PATCH CORRECTS A PROBLEM WHERE A PRINTER FILE DECLARED GREATER THAN 132 CHARACTERS WOULD HANG THE SYSTEM OR CAUSE SYSTEM PROBLEMS.

CHANGE NO. 108 (2 CARDS).

THIS WILL PUT ZEROES INSTEAD OF AN "0" AS A PADDING CHARACTER IN OUTPUT FROM A FORTRAN PROGRAM.

CHANGE NO. 109 (5 CARDS).

THIS CHANGE WILL ALLOW A COBOL68 USING/GIVING SORT TO WORK PROPERLY. WITHOUT THIS CHANGE, IF THE OUTPUT FILE IS THE SAME AS THE INPUT FILE, A SYSTEM HALT WOULD USUALLY RESULT. THIS WAS BECAUSE OF THE SPECIALIZED WAY COBOL68 FIBS ARE HANDLED.

CHANGE NO. 110 (014 CARDS).

THIS PATCH TO COBOLDECIMALTOOCTALCONVERT INTRINSIC CURRECTS A PROBLEM WHEREBY WHEN THE INTRINSIC WAS PASSED A FIELD FOR CONVERSION THAT CONTAINED ONLY NON-NUMERIC CHARACTERS THE CONVERSION WAS OFTEN DONE IMPROPERLY. FOR EXAMPLE! IF THE INTRINSIC WERE PASSED AN 11 DIGIT FIELD CONTAINING ALL "N"-S, THEN THE FIELD WAS CONVERTED TO THE NUMBER -55444444445 RATHER THAN TO THE NUMBER -55555555555.

CHANGE NO. 111 (11 CARDS).

THIS PATCH TO COBOLIODSK CORRECTS THE WAY SEEK-S AGAINST SEQUENTIAL OUTPUT DISK FILES ARE HANDLED. PRIOR TO THIS PATCH, USE OF A SEEK AGAINST A SERIAL OUTPUT FILE RESULTED IN AN INVALID PRL PROGRAM TERMINATION.

CHANGE NO. 112 (3 CARDS).

THIS PATCH TO COBOLIODSK CORRECTS THE WAY UNBLOCKED I-O RANDOM DISK FILES ARE HANDLED. PRIOR TO THIS PATCH, EACH WRITE HAD TO BE PRECEEDED BY A READ TO INSURE THAT ALL SUCEEDING I/O-S WERE DONE PROPERLY.

CHANGE NO. 201 (24 CARDS).

THIS PATCH IMPLEMENTS THE "QEND" FEATURE FOR THE INTRINSICS. SEE TSSMCP TEMPORARY PATCH # 213 ALSO A ZERO TIME OUT READ IS IMPLEMENTED.

CHANGE NO. 301 (244 CARDS).

THIS PATCH IMPLEMENTS THE INTRINSICS PORTION OF THE BASIC PRINT USING FEATURE.

ALGOL CHANGES.

CHANGE NO. 101 (2 CARDS).

THIS PATCH ELEMINATES A COMPILER LOOP CAUSED WHEN A FORMAL PARAMETER IN A PROCEDURE DECLARATION IS NOT INDICATED IN THE SPECIFICATION LIST.

CHANGE NO. 102 (1 CARD).

THIS PATCH ELEMINATES A COMPILER LOOP CAUSED WHEN THE FORMAL PARAMETER IN A PROCEDURE DECLARATION IS FOLLOWED BY A COMMA.

CHANGE NO. 103 (2 CARDS).

THIS PATCH CORRECTS AN INVALID INDEX CONDITION CAUSED WHEN TOO MANY USER OPTIONS HAVE BEEN SPECIFIED.

CHANGE NO. 104 (4 CARDS).

THIS PATCH CORRECTS A PROBLEM WHERE A PATCH CARD IS LOST WHEN BEGIN END PAIRS ARE NOT MATCHED AND PATCH CARD SEQUENCE NUMBERS ARE GREATER THAN THE SEQUENCE NUMBER OF THE "END." CARD IN THE SOURCE FILE.

CHANGE NO. 105 (13 CARDS).

THIS PATCH CORRECTS AN EOF NO LABEL ENCOUNTERED WHEN THE SOURCE "END." CARD IS PATCHED OVER AND THE PATCH DECK CONTAINS CARD SEQUENCE NUMBERS GREATER THAN THE SEQUENCE NUMBER OF THE "END." CARD IN THE SOURCE FILE.

CHANGE NO. 106 (38 CARDS).

THIS PATCH WILL ALLOW THE USER TO GET A DUMP OF THE COMPILER

GENERATED SEGMENT ZERO, PRT, SEGMENT DICT, FILE PARAMETER BLOCK. THIS IS CONTROLLED BY THE NEW OPTION "TABLES". IF THIS IS SET TRUE AT THE END OF A COMPILE THEN THESE TABLES WILL BE DUMPED AS THEY ARE WRITTEN TO THE CODE FILE. THIS OPTION WAS ADDED AS A COMPILER DEBUGGING AID.

CHANGE NO. 107 (139 CARDS).

THIS PATCH ALLOWS YOU THROUGH THE USE OF A $ OPTION TO INCLUDE SOURCE CODE ON THE DISK TO BE COMPILED INTO A USER PROGRAM.

THE SYNTAX FOR THE $ INCLUDE CARD IS:

$ INCLUDE <COPY PART> <FILE PART> <SEQUENCE PART>

<COPY PART> ::= <EMPTY> / + COPY

<FILE PART> ::= <MULTI-FILE ID>/<FILE ID> /
<MULTI-FILE ID>

<MULTI-FILE ID> ::= [ALPHANUMERIC STRING OF 7 OR FEWER CHARACTERS]

<FILE ID> ::= <EMPTY> / <ALPHANUMERIC STRING>

<SEQUENCE PART> ::= <STARTING SEQUENCE NUMBER> <ENDING SEQUENCE NUMBER> / <EMPTY>

<STARTING SEQUENCE NUMBER> ::= <UNSIGNED INTEGER>

<ENDING SEQUENCE NUMBER> ::= <EMPTY> / - <UNSIGNED INTEGER>

SOME EXAMPLES ARE:

```
$ INCLUDE A/B 1213-99932
$ INCLUDE A 12321-77651
$ INCLUDE+COPY SPECIAL/FILE 76333-124457
$ INCLUDE A 12223
$ INCLUDE A
$ INCLUDE + COPY IT
```

INCLUDE INSTRUCTS THE COMPILER TO COMPILE THE SOURCE CODE ON THE DISK FILE <FILE PART> OVER THE RANGE <SEQUENCE PART> AS PART OF THE ENTIRE PROGRAM. IN THIS MANNER, THE USER CAN COMPILE ALL OR PART OF AN AUXILLARY FILE(S) INTO HIS PROGRAM. IF THE <FILE ID> IS NOT PRESENT, THE USERCODE IS USED AS THE <FILE ID>.

THE STARTING AND ENDING SEQUENCE UNMBERS ARE INCLUSIVE. IF THE <SEQUENCE PART> IS EMPTY, THE ENTIRE FILE IS USED. IF ONLY THE STARTING SEQUENCE NUMBER IS PRESENT, THE FILE FROM THAT SEQUENCE NUMBER TO THE END OF THE FILE IS USED. IF BOTH SEQUENCE NUMBERS ARE PRESENT, THE FILE FROM THE STARTING SEQUENCE TO ENDING SEQUENCE, INCLUSIVE, IS USED. IF A NEW FILE IS BEING MADE, AND THE COPY PART IS EMPTY, ANY IMBEDDED $ INCLUDE CARDS WILL BE WRITTEN ON THE NEW FILE, BUT NOT THE INCLUDED FILES THEMSELVES. THIS PROVIDES THAT THE NEW FILE, WHEN IT ITSELF IS COMPILED, WILL INCLUDE THE FILES, WHILE AT THE SAME TIME ALLOWING THE INCLUDED FILES TO BE UPDATED INDEPENDENTLY OF THE NEW FILE.

IF A NEW FILE IS BEING MADE AND THE COPY PART IS PRESENT, THE IMBEDDED $ INCLUDE CARDS WILL NOT BE WRITTEN OUT ON THE NEW FILE, BUT RATHER THE INCLUDED RECORDS THEMSELVES WILL BE COPIED ONTO THE NEW FILE. THE COPY PART IS IGNORED IF A NEW FILE IS NOT BEING MADE. NOTE THAT INCLUDED FILES CAN HAVE $ INCLUDE CARDS IMBEDDED WITHIN THEM, AND THUS RECURSION ON THE $ INCLUDE CARDS CAN OCCUR.

CHANGE NO. 108 (17 CARDS).

THIS PATCH MAKES IT POSSIBLE TO SET OR RESET COMPILER OPTION "SEQXEQ" ANY NUMBER OF TIMES BEFORE THE FIRST BEGIN. AFTER THAT ITS CONDITION WILL REMAIN UNCHANGED. IT STILL MAY NOT BE POPPED (POP WILL ACT AS A RESET).

BASIC CHANGES.

CHANGE NO. 101 (14 CARDS).

THIS PATCH ELIMINATES AN INVALID EOI ENCOUNTERED WHEN TAKING AN EOF BRANCH ON AN INPUT STATEMENT WHILE IN A GO SUB.

CHANGE NO. 301 (122 CARDS).

THIS PATCH IMPLEMENTS THE PRINT-USING FEATURE.

COBOL CHANGES.

CHANGE NO. 101 (1 CARD).

THIS CHANGE FLAGS THE CONSTRUCT "PICTURE IS S" AS A FATAL SYNTAX ERROR.

CHANGE NO. 102 (10 CARDS).

THIS CHANGE CORRECTS A STACK OVERFLOW FROM OCCURRING WHEN MOVING TO COMP TABLES.

CHANGE NO. 103 (1 CARD).

THIS PATCH CORRECTS THE CODE GENERATED FOR THE INVALID KEY SYNTAX WHEN WRITING SEQUENTIAL DISK FILES THAT CONTAIN AN ACTUAL KEY CLAUSE IN THE FILE DESCRIPTION.

COBOL68 CHANGES.

CHANGE NO. 101 (31 CARDS).

THIS PATCH CORRECTS ALL KNOWN PROBLEMS WITH DISPLAYING SIGNED NUMERICS. PRIOR TO THIS PATCH, THE SIGN WAS LOST WHEN DISPLAYING LEFT-SIGNED FIELDS (I.E., S OR V). WHEN THE CONSTRUCT:

DISPLAY NUMERIC-ITEM "QUOTED STRING"

WAS USED, THE QUOTED-STRING WAS ALSO LOST. IN ADDITION, THIS PATCH CORRECTS THE PROBLEMS ASSOCIATED WITH RIGHT-SIGNED FIELDS (I.E., +, -, CR, AND DB). PRIOR TO THIS PATCH, THE SIGN WAS LOST WHENEVER A SIGNED-NUMERIC WAS MOVED TO A RIGHT-SIGNED FIELD.

CHANGE NO. 102 (1 CARD).

THIS PATCH CAUSES A SYNTAX ERROR TO BE GENERATED WHENEVER THE CONSTRUCT: WRITE FROM "QUOTED STRING" IS USED. PRIOR TO THIS PATCH, A SYNTAX ERROR WAS NOT EMITTED, HOWEVER, THE CODE GENERATED OFTEN CAUSED A SYSTEM HALT.

CHANGE NO. 104 (6 CARDS).

THIS PATCH CORRECTS A PROBLEM WITH B-INSERTION EDITTING. PRIOR TO THIS PATCH, IF AN ELEMENTARY ITEM HAD A PICTURE OF ONLY B-S, THEN THE LENGTH OF THAT ELEMENTARY ITEM WAS NOT COUNTED IN THE LENGTH OF THE GROUP TO WHICH IT BELONGED. THIS CONDITION COULD CAUSE AN INVALID LINK.

CHANGE NO. 105 (2 CARDS).

THIS PATCH CORRECTS PROBLEMS WITH THE CONSTRUCT: CLOSE WITH CRUNCH.

CHANGE NO. 106 (8 CARDS).

THIS PATCH ASSURES THAT THE CLAUSE "ASSIGN TO SORT DISK" GOES ONLY

TO DISK AND NOT TO TAPE. PREVIOUSLY, IF THIS CLAUSE WAS USED, THE SORT WENT TO TAPE IF NOT ENOUGH DISK WAS AVAILABLE.

CHANGE NO. 107 (5 CARDS).

THIS PATCH FIXES THE VALUE CLAUSE SO THAT IF THE NUMBER OF CHARACTERS WITHIN THE QUOTES IS GREATER THAN 63 CHARACTERS THE ENTIRE QUOTED STRING IS PASSED AND NOT JUST 63 CHARACTERS. THIS PATCH ALSO APPLIES TO FIGURATIVE CONSTANTS GREATER THAN 63 CHARACTERS.

CHANGE NO. 108 (1 CARD).

THIS PATCH CORRECTS THE CODE FOR THE "SEARCH" STATEMENT SO THAT IT WILL SYNTAX PROPERLY. PREVIOUSLY, THE "WHEN" PORTION REQUIRED A SEMI-COLON PRECEDING IT.

CHANGE NO. 109 (1 CARD).

PRIOR TO THIS PATCH, ANY STATEMENT WITH A "D" IN COLUMN 7 WAS NOT COMPILED IN WHEN "WITH DEBUGGING MODE" WAS SPECIFIED.

CHANGE NO. 110 (2 CARDS).

THIS PATCH CORRECTS A PROBLEM WITH SUBSCRIPTING. IF A SUBSCRIPT IS AN ARITHMETIC EXPRESSION (I.E., TABLES(SUB + 1)) AND "SUB" IS A DATANAME DESCRIBED WITH A PICTURE OF 9(11) COMP OR COMP-1, THEN A DOUBLE PRECISION FIELD IS GENERATED TO DO THE ADD. THE SECOND WORD WAS BEING LEFT IN THE STACK THUS CAUSING VARIOUS DIFFERENT PROBLEMS, DEPENDING ON THE PROGRAM CONTENT. IN SOME CASES, A SYSTEM HANG RESULTED.

CHANGE NO. 112 (4 CARDS).

THIS PATCH CORRECTS CERTAIN DISCREPANCIES BETWEEN COBOL68 AND B6700 COBOL IN SYNTAXING AND GENERATING CODE FOR THE CLOSE STATEMENT. WHEN CLOSING A FILE ASSIGNED TO A CARD-READER, A CARD-PUNCH, OR TO A PRINTER, ONLY THE OPTIONS "WITH RELEASE" AND "WITH LOCK" ARE ALLOWED.

HOWEVER, THESE ARE IGNORED AND THE ACTION TAKEN IS THE SAME AS FOR A SIMPLE CLOSE:

(1) THE INPUT/OUTPUT AREAS ARE RELEASED.
(2) THE TRAILER LABEL(IF ANY) IS WRITTEN.
(3) THE UNIT IS RETURNED TO THE MCP.

CHANGE NO. 113 (1 CARD).

THIS PATCH CORRECTS THE CODE EMITTED FOR STATEMENTS WHICH MOVE A NUMERIC ITEM OF 12 OR MORE DIGITS TO AN EDITTED NUMERIC ITEM DECLARED WITH COMPLETE ZERO SUPPRESSION. IN ADDITION, THIS PATCH CORRECTS THE CODE EMITTED FOR STATEMENTS WHICH COMPUTE A VALUE AND THEN STORE THAT VALUE IN AN EDITTED NUMERIC FIELD DECLARED WITH COMPLETE ZERO SUPPRESSION, FOR EXAMPLE:

```
01  REC.
    03  FLD1     PIC 9(9).
    03  FLD2     PIC 9(9).
    03  FLD3     PIC 9(9).
    03  FLD4     PIC 9(9).
    03  FLD5     PIC 9(9).
    03  Z-FLD    PIC Z(9).
ADD FLD1 FLD2 FLD3 FLD4 FLD5
    GIVING Z-FLD.
```

PRIOR TO THIS PATCH, AN EXTRA "DEL" WAS EMITTED WHICH COULD CAUSE AN INVALID ADDRESS PROGRAM TERMINATION, A STACK OVERFLOW PROGRAM TERMINATION, AN INVALID LINK, OR SOME OTHER UNPREDICTABLE AND UNEXPLAINABLE RESULT.

CHANGE NO. 114 (24 CARDS).

THIS PATCH IMPLIMENTS THE NEW DOLLAR-CARD OPTION TSSCOPY. LIKE ALL OTHER DOLLAR-CARD OPTIONS, IT MAY BE SET, RESET, AND POPPED.

TSSCOPY SET:
THE INCOMING LIBRARY-FILE INPUT RECORDS ARE EXPECTED TO BE IN CANDE FORMAT:
COLUMNS 1 THRU 72 - COBOL STATEMENT
COLUMNS 73 THRU 80 - SEQUENCE NUMBER(ONLY 75 THRU 80 ARE SIGNIFICANT)

TSSCOPY RESET:
THE INCOMING LIBRARY-FILE INPUT RECORDS ARE EXPECTED TO BE IN STANDARD COBOL-STATEMENT FORMAT:

COLUMNS 1 THRU 6 - SEQUENCE NUMBER
COLUMNS 8 THRU 72 - COBOL STATEMENT
COLUMNS 73 THRU 80 - IDENTIFICATION, COMMENT, OR BLANK

BY DEFAULT, TSSCOPY IS SET WHEN COMPILING FROM REMOTE THROUGH CANDE. IT IS RESET WHEN COMPILING IN A BATCH MODE FROM THE CARD READER.

PRIOR TO THIS PATCH, COPIED FILES WERE EXPECTED TO BE IN THE SAME FORMAT AS THE PRIMARY INPUT FILE (CARD). CONSEQUENTLY, BATCH COMPILATIONS WERE UNABLE TO COPY CANDE FORMAT LIBRARY FILE AND REMOTE COMPILATIONS WERE UNABLE TO COPY STANDARD COBOL-STATEMENT FORMAT LIBRARY FILES. THIS PATCH ALSO CORRECTS 1 PROBLEM WITH THE SEQUENCE-RANGE OPTION OF THE COPY STATEMENT. PRIOR TO THIS PATCH, THE FIRST SIX COLUMNS OF THE INCOMING LIBRARY-FILE RECORD WERE ALWAYS USED FOR THE SEQUENCE- RANGE COMPARISON--EVEN IF THE INCOMING LIBRARY-FILE RECORD WAS IN CANDE FORMAT.

CHANGE NO. 115 (1 CARD).

THIS PATCH CORRECTS AN INFINITE-LOOP CONDITION IN PASS-1 OF A COBOL68 COMPILE CAUSED BY A MISSING LEVEL NUMBER IN THE DATA DIVISION.

CHANGE NO. 116 (16 CARDS).

THIS PATCH CORRECTS THE CODE EMITTED FOR THE PERFORM VERB, OPTION 4, WHEN VARYING FROM "FORMULA-1". PRIOR TO THIS PATCH, USE OF THIS CONSTRUCT RESULTED IN A STACK OVERFLOW.

CHANGE NO. 117 (1 CARD).

THIS PATCH ASSURES THAT A SYNTAX ERROR IS GENERATED WHENEVER A DISK FILE IS DECLARED AS OPTIONAL. PRIOR TO THIS PATCH, A SYNTAX ERROR WAS NOT GENERATED; AT RUN TIME, HOWEVER, AN "OF" INPUT MESSAGE WAS NOT ACCEPTABLE.

CHANGE NO. 118 (16 CARDS).

THIS PATCH CORRECTS THE PROBLEMS ASSOCIATED WITH NESTED REDEFINES.
PRIOR TO THIS PATCH, THE LENGTH OF THE REDEFINED ELEMENT WAS
CALCULATED INCORRECTLY. FOR EXAMPLE:

```
... 03  GROUP-ITEM           SIZE 96.
        05  ANY-ITEM         PIC X(18).
        05  ...
        .
        .
        .
    03  REDF-GROUP-ITEM REDEFINES GROUP-ITEM.
        05  ANOTHER-ITEM     PIC 9(6).
        05  ...
        .
        .
        .
        05  ELEM-ITEM        PIC Z(5).
        05  REDF-ELEM-ITEM REDEFINES ELEM-ITEM PIC X(5).
        05  ...
        .
        .
        .
```

THE LENGTH OF ELEM-ITEM WAS CALCULATED TO BE 96 - THE LENGTH OF
GROUP-ITEM AND REDF-GROUP-ITEM - NOT 5, ITS CORRECT LENGTH.

THIS IN TURN CAUSED PROBLEMS THROUGHOUT THE PROCEDURE DIVISION,
PARTICULARILY WHEN THE REDEFINED ITEM WAS THE RECEIVING FIELD IN A
MOVE STATEMENT. A SYSTEM HALT COULD RESULT DUE TO OVER-WRITTEN
MEMORY LINKS.

THIS PATCH ALSO CAUSES A NON-FATAL WARNING MESSAGE TO BE EMITTED
WHENEVER AN 01 LEVEL REDEFINITION IS NOT THE SAME LENGTH AS THE 01
LEVEL AREA THAT IT REDEFINES. PRIOR TO THIS PATCH, THAT ERROR WENT
UNFLAGGED.

CHANGE NO. 119 (3 CARDS).

THIS PATCH CORRECTS PROBLEMS WITH ZERO-INSERTION EDITTING. PRIOR TO
THIS PATCH, WHENEVER A NUMERIC ITEM WAS MOVED TO A ZERO-INSERTION
EDITTED ITEM, THE NUMBER WAS OFFSET TO THE RIGHT A NUMBER OF DIGITS
EQUAL TO THE NUMBER OF ZEROES TO BE INSERTED.

CHANGE NO. 120 (1 CARD).

THIS PATCH CORRECTS A PROBLEM WHEREBY THE ERROR "MISSING PROGRAM" IS NOT BEING FLAGGED AS FATAL. THIS CAUSES PROBLEMS IN COMPILE-AND-GO SITUATIONS. IN PARTICULAR, IF "MISSING PROGRAM" IS THE ONLY ERROR, THEN THE COMPILE PHASE GOES TO A NORMAL EOJ, NOT A SYNTAX ERROR EOJ. WHEN THE MCP TRIES TO FIRE OFF THE GO PHASE, ESPDISK ERRORS RESULT.

CHANGE NO. 121 (1 CARD).

THIS PATCH FORCES THE CONSTRUCT: "IF NUMERIC" TO FUNCTION PROPERLY WHEN TESTING A SIGNED-NUMERIC FIELD. PRIOR TO THIS PATCH, IF THE FIELD CONTAINED A NEGATIVE VALUE, THE TEST RETURNED A VALUE OF FALSE.

CHANGE NO. 122 (3 CARDS).

THIS PATCH CAUSES A NON-FATAL WARNING MESSAGE TO BE EMITTED WHENEVER THE LENGTH OF AN ALPHA-NUMERIC LITERAL IN A VALUE CLAUSE EXCEEDS THE DECLARED SIZE OF THE ITEM.

CHANGE NO. 123 (30 CARDS).

THIS PATCH PREVENTS A SYSTEM HANG FROM OCCURING WHEN A COBOL68 PROGRAM FAILS TO TERMINATE WITH AN INVALID EOJ BECAUSE THE LAST STATEMENT COMPILED EXCEEDED THE SEGMENT LIMIT AND THE PROGRAMMER DID NOT INCLUDE A FINAL LABEL, FOR INSTANCE, END-OF-JOB, TO WHICH HE NEVER BRANCHES AND WHICH IS FOLLOWED BY NO STATEMENTS.

CHANGE NO. 125 (30 CARDS).

THIS PATCH CORRECTS THE CODE EMITTED FOR LONG, NESTED IF STATEMENTS, THAT IS, IF STATEMENTS THAT NECESSITATE GENERATION OF MORE THAN 256 WORDS OF CODE. WITHOUT THIS PATCH, THE COMPILER WILL GENERATE CODE THAT WHEN EXECUTED MAY CAUSE PROGRAM MALFUNCTION, PROGRAM ABORT OR A SYSTEM HALT.

CHANGE NO. 126 (1 CARD).

THIS PATCH CORRECTS MANY PROBLEMS ASSOCIATED WITH THE USE OF 88-

LEVEL CONDITIONALS. PRIOR TO THIS PATCH, AN ENTRY WAS NOT REMOVED FROM A COMPILER TABLE WHEN CODE GENERATION FOR A CONDITIONAL WAS COMPLETED. CONSEQUENTLY, CODE GENERATION FOR THE REMAINDER OF THE CONDITIONAL STATEMENT MAY HAVE BEEN INCORRECTLY DONE.

CHANGE NO. 127 (2 CARDS).

THIS PATCH CORRECTS THE CODE EMITTED FOR 88-LEVEL CONDITIONALS ASSOCIATED WITH 77-LEVEL ITEMS. PRIOR TO THIS PATCH, A VALUE WAS INCORRECTLY COMPUTED AND INSERTED INTO A COMPILER TABLE DURING PASS-1 SYNTAXING. DURING PASS-2, THIS INCORRECTLY COMPUTED VALUE CAUSED THE CODE GENERATION FOR THE CONDITIONAL TO BE DONE IMPROPERLY. EXECUTION OF THIS INCORRECT CODE COULD RESULT IN AN INVALID INDEX PROGRAM ABORT BUT MORE OFTEN THAN NOT SIMPLY RESULTED IN PROGRAM MALFUNCTION --- A CONDITION OFTEN BLAMED ON A PROGRAM LOGIC ERROR.

CHANGE NO. 128 (1 CARD).

THIS PATCH CORRECTS PROBLEMS ASSOCIATED WITH FLOATING $ SIGN, + SIGN, AND - SIGN EDITTING.

CHANGE NO. 129 (1 CARD).

THIS PATCH CORRECTS THE CODE EMITTED FOR THE CONSTRUCT: MOVE <DATA-NAME-1> TO <DATA-NAME-2> <DATA-NAME-3> ... PRIOR TO THIS PATCH, DATA-NAME-1 WAS OFTEN ONLY MOVED TO THE FIRST RECEIVING FIELD.

CHANGE NO. 130 (15 CARDS).

THIS PATCH CORRECTS THE CODE GENERATED TO EDIT NUMERIC FIELDS DECLARED EITHER WITH COMPLETE ZERO SUPPRESSION OR EXPLICITLY "BLANK WHEN ZERO". PRIOR TO THIS PATCH, THE SENDING FIELD WAS TREATED AS A SINGLE PRECISION OPERAND WHEN TESTING FOR ZERO --- EVEN IF THE SENDING FIELD WAS IN FACT A DOUBLE PRECISION OPERAND.

CHANGE NO. 131 (2 CARDS).

THIS PATCH CORRECTS THE CODE EMITTED FOR COMPOUND-CONDITION IF

STATEMENTS. PRIOR TO THIS PATCH IF THE CODE FOR SUCH STATEMENTS WERE GENERATED IN THE ADDRESS RANGE 51210 THRU 102313 OF A CODE SEGMENT THE CODE WAS OFTEN INCORRECT, PARTICULARILY IF THE COMPOUND-CONDITION CONTAINED "NOT" LOGIC. EXECUTION OF THIS INCORRECTLY COMPILED CODE COULD RESULT IN PROGRAM ABORT OR SYSTEM HALT.

CHANGE NO. 201 (3 CARDS).

THIS PATCH WILL CAUSE THE CORE ESTIMATE OF ALL PROGRAMS COMPILED TO INCLUDE THE SORT MEMORY SIZE SPECIFIED IN THE OBJECT-COMPUTER CLAUSE.

ESPOL CHANGES.

CHANGE NO. 101 (2 CARDS).

THIS PATCH ELEMINATES A COMPILER LOOP CAUSED WHEN A FORMAL PARAMETER IN A PROCEDURE DECLARATION IS NOT INDICATED IN THE SPECIFICATION LIST.

CHANGE NO. 102 (1 CARD).

THIS PATCH ELEMINATES A COMPILER LOOP CAUSED WHEN THE FORMAL PARAMETER IN A PROCEDURE DECLARATION IS FOLLOWED BY A COMMA.

CHANGE NO. 103 (2 CARDS).

THIS PATCH CORRECTS AN INVALID INDEX CONDITION CAUSED WHEN TOO MANY USER OPTIONS HAVE BEEN SPECIFIED.

CHANGE NO. 104 (4 CARDS).

THIS PATCH CORRECTS A PROBLEM WHERE A PATCH CARD IS LOST WHEN BEGIN END PAIRS ARE NOT MATCHED AND PATCH CARD SEQUENCE NUMBERS ARE GREATER THAN THE SEQUENCE NUMBER OF THE "END." CARD IN THE SOURCE FILE.

CHANGE NO. 105 (11 CARDS).

THIS PATCH CORRECTS AN EOF NO LABEL ENCOUNTERED WHEN THE SOURCE "END." CARD IS PATCHED OVER AND THE PATCH DECK CONTAINS CARD SEQUENCE NUMBERS GREATER THAN THE SEQUENCE NUMBER OF THE "END." CARD IN THE SOURCE FILE.

FORTRAN CHANGES.

CHANGE NO. 101 (1 CARD).

THIS PATCH CORRECTS AN FORMAT ERROR IF THE ONLY LIST ITEM REQUIRED WAS FOR A VARIABLE FORMAT PHRASE.

CHANGE NO. 102 (2 CARDS).

THIS PATCH WILL ALLOW THE NEW TAPE FILE TO HAVE THE SAME MFID AND FID AS THE TAPE FILE.

CHANGE NO. 103 (7 CARDS).

THIS PATCH WILL ALLOW THE USE OF COMPLEX EXPRESSIONS IN THE LOGICAL IF STATEMENT AND IN LOGICAL STATEMENTS IN GENERAL. THE .EQ. AND .NE. OPERATORS CAN NOW BE USED TO COMPARE TWO LOGICAL ITEMS.

SYNTAX EXAMPLES:

```
COMPLEX C1,C2
LOGICAL L1
IF(C1 .EQ. C2) GO TO 1000
L1 = C1 .NE. C2
```

CHANGE NO. 104 (1 CARD).

THIS PATCH WILL PREVENT THE FORTRAN COMPILER FROM BEING DS-ED DUE TO AN INVALID INDEX WHEN THE COMPILING PROGRAM REFERS TO A VARIABLE IN AN INCONSISTENT MANNER. THE COMPILER WILL NOW FLAG THE CONDITION WITH AN ERROR MESSAGE.

CHANGE NO. 105 (2 CARDS).

THIS PATCH CORRECTS AN ERROR THAT OCCURS IF A COMPLEX OR DOUBLE PRECISION FUNCTION IS PASSED AS A PARAMETER TO A FUNCTION OR SUBROUTINE. THE PROGRAM COULD GET DS-ED FOR INVALID INDEX OR STACK OVERFLOW AND SOMETIMES SYSTEM HANGS COULD OCCUR.

CHANGE NO. 106 (2 CARDS).

THIS PATCH ELIMINATES ERRONEOUS DATA ON THE SYMBOL FILE

CHANGE NO. 107 (1 CARD).

THIS PATCH WILL CORRECT A PROBLEM WITH "$ INCLUDE" NOT WORKING.

CHANGE NO. 108 (1 CARD).

THIS CHANGE WILL ALLOW THE FORTRAN COMPILER TO CORRECTLY SYNTAX CALLS ON FUNCTION STATEMENTS. WITHOUT THIS PATCH THE COMPILER WOULD NOT GIVE A SYNTAX ERROR IF THE ACTUAL PARAMETER IS A DOUBLE PRECISION VARIABLE AND THE FORMAL PARAMETER IS A REAL VARIABLE, THUS THE STACK WOULD BE SET UP WRONG AND THE OBJECT PROGRAM COULD BE DUMPED FOR INVALID ADDRESS OR STACK OVERFLOW, AND SOMETIMES SYSTEM HALTS COULD HAPPEN. ALSO IF A REAL VARIABLE WERE PASSED TO A INTEGER FORMAL PARAMETER AN INVALID SYNTAX ERROR WOULD BE GIVEN.

NOTE: THE FORMAL PARAMETERS, SOMETIMES CALLED DUMMY VARIABLES CAN BE GIVEN A TYPE, OTHER THAN THE DEFAULT TYPE, BY REFERENCE TO THEM IN A TYPE DECLARATIVE STATEMENT OR AN IMPLICIT TYPE STATEMENT.

EXAMPLE:

```
REAL *8 X,F, B      IN THIS EXAMPLE THE DOUBLE PRECISION
F(X) = X*X        FUNCTION F HAS ONE DOUBLE PRECISION
A = SNGL(F(B))    PARAMETER B.
STOP
END
```

XALGOL CHANGES.

CHANGE NO. 101 (2 CARDS).

THIS PATCH ELEMINATES A COMPILER LOOP CAUSED WHEN A FORMAL PARAMETER IN A PROCEDURE DECLARATION IS NOT INDICATED IN THE SPECIFICATION LIST.

CHANGE NO. 102 (1 CARD).

THIS PATCH ELEMINATES A COMPILER LOOP CAUSED WHEN THE FORMAL PARAMETER IN A PROCEDURE DECLARATION IS FOLLOWED BY A COMMA.

CHANGE NO. 103 (2 CARDS).

THIS PATCH CORRECTS AN INVALID INDEX CONDITION CAUSED WHEN TOO MANY USER OPTIONS HAVE BEEN SPECIFIED.

CHANGE NO. 104 (4 CARDS).

THIS PATCH CORRECTS A PROBLEM WHERE A PATCH CARD IS LOST WHEN BEGIN END PAIRS ARE NOT MATCHED AND PATCH CARD SEQUENCE NUMBERS ARE GREATER THAN THE SEQUENCE NUMBER OF THE "END."CARD IN THE SOURCE FILE.

CHANGE NO. 105 (13 CARDS).

THIS PATCH CORRECTS AN EOF NO LABEL ENCOUNTERED WHEN THE SOURCE "END." CARD IS PATCHED OVER AND THE PATCH DECK CONTAINS CARD SEQUENCE NUMBERS GREATER THAN THE SEQUENCE NUMBER OF THE "END." CARD IN THE SOURCE FILE.

CHANGE NO. 106 (2 CARDS).

THIS PATCH CHANGES THE CORE ESTIMATE GENERATED BY THE COMPILER TO

INCLUDE THE SIZE OF THE FILE BUFFERS. WITHOUT THIS PATCH THE BUFFER SIZE IS COMPUTED TO BE ZERO.

CHANGE NO. 107 (38 CARDS).

THIS PATCH WILL ALLOW THE USER TO GET A DUMP OF THE COMPILER GENERATED SEGMENT ZERO, PBT, SEGMENT DICT, FILE PARAMETER BLOCK. THIS IS CONTROLLED BY THE NEW OPTION "TABLES". IF THIS IS SET TRUE AT THE END OF A COMPILE THEN THESE TABLES WILL BE DUMPED AS THEY ARE WRITTEN TO THE CODE FILE. THIS OPTION WAS ADDED AS A COMPILER DEBUGGING AID.

CHANGE NO. 108 (139 CARDS).

THIS PATCH ALLOWS YOU THROUGH THE USE OF A $ OPTION TO INCLUDE SOURCE CODE ON THE DISK TO BE COMPILED INTO A USER PROGRAM.

THE SYNTAX FOR THE $ INCLUDE CARD IS:

$ INCLUDE <COPY PART> <FILE PART> <SEQUENCE PART>

<COPY PART> ::= <EMPTY> / + COPY

<FILE PART> ::= <MULTI-FILE ID>/<FILE ID> /
<MULTI-FILE ID>

<MULTI-FILE ID> ::= [ALPHANUMERIC STRING OF 7 OR FEWER CHARACTERS]

<FILE ID> ::= <EMPTY> / <ALPHANUMERIC STRING>

<SEQUENCE PART> ::= <STARTING SEQUENCE NUMBER> <ENDING SEQUENCE NUMBER> / <EMPTY>

<STARTING SEQUENCE NUMBER> ::= <UNSIGNED INTEGER>

<ENDING SEQUENCE NUMBER> ::= <EMPTY> / - <UNSIGNED INTEGER>

SOME EXAMPLES ARE:

```
$ INCLUDE A/B 1213-99932
$ INCLUDE A 12321-77651
$ INCLUDE+COPY SPECIAL/FILE 76333-124457
$ INCLUDE A 12223
$ INCLUDE A
```

$ INCLUDE + COPY IT

INCLUDE INSTRUCTS THE COMPILER TO COMPILE THE SOURCE CODE ON THE DISK FILE <FILE PART> OVER THE RANGE <SEQUENCE PART> AS PART OF THE ENTIRE PROGRAM. IN THIS MANNER, THE USER CAN COMPILE ALL OR PART OF AN AUXILLARY FILE(S) INTO HIS PROGRAM. IF THE <FILE ID> IS NOT PRESENT, THE USERCODE IS USED AS THE <FILE ID>.

THE STARTING AND ENDING SEQUENCE UNMBERS ARE INCLUSIVE. IF THE <SEQUENCE PART> IS EMPTY, THE ENTIRE FILE IS USED. IF ONLY THE STARTING SEQUENCE NUMBER IS PRESENT, THE FILE FROM THAT SEQUENCE NUMBER TO THE END OF THE FILE IS USED. IF BOTH SEQUENCE NUMBERS ARE PRESENT, THE FILE FROM THE STARTING SEQUENCE TO ENDING SEQUENCE, INCLUSIVE, IS USED. IF A NEW FILE IS BEING MADE, AND THE COPY PART IS EMPTY, ANY IMBEDDED $ INCLUDE CARDS WILL BE WRITTEN ON THE NEW FILE, BUT NOT THE INCLUDED FILES THEMSELVES. THIS PROVIDES THAT THE NEW FILE, WHEN IT ITSELF IS COMPILED, WILL INCLUDE THE FILES, WHILE AT THE SAME TIME ALLOWING THE INCLUDED FILES TO BE UPDATED INDEPENDENTLY OF THE NEW FILE.

IF A NEW FILE IS BEING MADE AND THE COPY PART IS PRESENT, THE IMBEDDED $ INCLUDE CARDS WILL NOT BE WRITTEN OUT ON THE NEW FILE, BUT RATHER THE INCLUDED RECORDS THEMSELVES WILL BE COPIED ONTO THE NEW FILE. THE COPY PART IS IGNORED IF A NEW FILE IS NOT BEING MADE. NOTE THAT INCLUDED FILES CAN HAVE $ INCLUDE CARDS IMBEDDED WITHIN THEM, AND THUS RECURSION ON THE $ INCLUDE CARDS CAN OCCUR.

CHANGE NO. 109 (17 CARDS).

THIS PATCH MAKES IT POSSIBLE TO SET OR RESET COMPILER OPTION "SEQXEQ" ANY NUMBER OF TIMES BEFORE THE FIRST BEGIN. AFTER THAT ITS CONDITION WILL REMAIN UNCHANGED. IT STILL MAY NOT BE POPPED (POP WILL ACT AS A RESET).

CANDE CHANGES.

CHANGE NO. 101 (1 CARD).

THIS PATCH CORRECTS AN ERROR THAT WOULD ALLOW TWO OR MORE JOBS TO BE ASSIGNED TO A TERMINAL AT THE SAME TIME. THE ERROR WOULD OCCUR IF AN INVALID "TO" MESSAGE WAS ENTERED.

CHANGE NO. 102 (2 CARDS).

IF THE QMARK TO [USERCODE] CONSTRUCT IS USED AND THE USERCODE CONTAINS AN ALPHANUMERIC CHARACTER, THE MESSAGE WILL NOT BE SENT, AND A "NOT ON" WILL BE THE REPLY.

CHANGE NO. 103 (1 CARD).

IF A MAKE IS ENTERED AND A DUPLICATE FILE OCCURS SEVERAL BAD THINGS HAPPEN. IF THE NEXT COMMAND IS A LOAD THEN THE FILE IS OK FOR A COPY TO PRINTER BUT A PRINT WILL GET A NO FILE.

CHANGE NO. 104 (1 CARD).

THIS PATCH WILL CORRECT AN ERROR THAT COULD CAUSE CANDE TO BE DS-ED FOR AN INVALID INDEX. THE ERROR HAPPENS IF MANY JOBS ARE "CHAIN"ED. THREE WORDS WERE LEFT IN THE PSEUDO STACK FOE EACH CHAIN REQUEST FOR A LINE, AND THUS AFTER SEVERAL JOBS WERE RUN WITHOUT AN EOJ THAT DID NOT REQUEST A CHAIN, CANDE WOULD DO AN INVALID INDEX.

COOL CHANGES.

CHANGE NO. 101 (9 CARDS).

THIS PATCH WILL NOT ALLOW A FILE TO BE CREATED DURING A COLD START THAT HAS A DISK ADDRESS OF LESS THAN DIRECT + 4.

CHANGE NO. 103 (018 CARDS).

THIS PATCH STRAIGHTNS OUT THE MESS PRODUCED WHEN COOL STARTING AND THE OPERATOR DOES NOT WANT TO REMOVE PSEUDO DECKS ON DISK. PREVIOUSLY STRANGE THINGS HAPPENED WHEN THE SYSTEM WAS TOLD NOT TO REMOVE THE DECKS. THEY COULD NOT BE RUN NOR REMOVED; THEY ALSO CAUSED DUP LIBRARY CONDITIONS FOR LOAD/CONTROL.

CHANGE NO. 104 (009 CARDS).

THIS PATCH ALLOWS COOL/START TO DETERMINE IF ANYTHING WAS ACTUALLY WRITTEN ON THE LINE PRINTERS BEFORE HE SPOUTS THE MESSAGE "CHECK PRINTER FOR OUTPUT".

CHANGE NO. 105 (001 CARD).

THIS PATCH RESETS ALL OF THE OPTIONS IN THE MCP SO THAT ANY OPTIONS CAN BE SET WHEN COOL STARTING.

DCFILL CHANGES.

CHANGE NO. 101 (1 CARD).

THIS PATCH PUTS THE CORRECT MCP LEVEL INTO PRT/SAVE SO THAT DUMPANL/
UTILITY WILL FIND THE RIGHT LEVEL WHEN HE LOOKS OUT ON DISK.

DUMP CHANGES.

CHANGE NO. 101 (001 CARD).

THIS PATCH ALLOWS DUMPANL/UTILITY TO PRINT THE CORRECT LEVEL AND SUBLEVEL OF THE MCP/DISK BY READING IN THE CORRECT VALUE FROM PRT/SAVE.

CHANGE NO. 102 (010 CARDS).

THIS PATCH PRINTS OUT THE READQUE.

MAKCAST CHANGES.

CHANGE NO. 101 (38 CARDS).

THIS PATCH IMPLEMENTS AN OPTINOAL SEQUENCE CHECK OF INPUT CARDS.

ROTO CHANGES.

CHANGE NO. 101 (011 CARDS).

THIS PATCH LISTS THE I/O CHANNELS USED.

TSDUMP CHANGES.

CHANGE NO. 101 (001 CARD).

THIS PATCH ALLOWS TSDUMP/ANALYZE TO PRINT THE CORRECT LEVEL AND SUBLEVEL OF THE TSS/MCP BY READING IN THE CORRECT VALUES FROM TSS/PRT.

TSFILL CHANGES.

CHANGE NO. 101 (1 CARD).

THIS PATCH PUTS THE CORRECT MCP LEVEL INTO TSS/PRT SO THAT TSDUMP/ANALYZE WILL FIND THE RIGHT LEVEL WHEN HE LOOKS OUT ON DISK.

APPENDIX A - NEW LIBRARY MAINTENANCE

DOCUMENTATION FOR THE NEW
B5700 LIBRARY MAINTENANCE

OPERATIONAL DESCRIPTION

B5700 LIBRARY MAINTENANCE HAS BEEN EXTENDED TO INCLUDE CERTAIN FEATURES ASSOCIATED WITH THE B6700 "COPY" STATEMENT AND MORE, IT IS NOW POSSIBLE TO MODIFY EACH FILE NAME PAIR, GIVEN IN THE CONTROL CARD NAME LIST, BY AN "EXCEPT" LIST (A LIST OF NAMES RELATED TO THE PRIMARY NAME WHICH THE USER WISHES EXCLUDED FROM THE ACTIONS TAKEN ON THE PRIMARY) AND/OR AN "AS" CLAUSE (A MEANS OF CHANGING THE NAME OF THE PRIMARY ON THE DESTINATION). USERS MAY ALSO SPECIFY A NUMBER OF INPUT SOURCES WITH ONE DESTINATION, ANY MODE OF TRANSFER INVOLVING TAPES AND DISK IS AVAILABLE, EXCLUDING DISK TO DISK. CONTROL OPTIONS MAY BE SPECIFIED FOR EACH SOURCE AND A USER SPECIFIED MAXIMUM NUMBER OF FILES PER OUTPUT UNIT MAY BE DECLARED. IN ORDER TO MAKE USE OF MULTIPLE SOURCE CONTROL CARDS ONE MUST USE THE NEW "COPY" CONTROL CARD.

THE REMOVE CONTROL CARD HAS BEEN ALTERED TO MAKE USE OF THE "EXCEPT" LIST FACILITY AND OLD LIBRARY MAINTENANCE CONTROL CARDS (ADD, DUMP, ETC.) MAY USE BOTH "EXCEPT" LISTS AND "AS" CLAUSES.

THE FOLLOWING ARE EXAMPLES OF THE USE OF "EXCEPT" AND "AS" MODIFIERS PERTAINING TO VARIOUS PRIMARY FILE NAMES WITHIN A LIST. (USE OF THE NAMES "A","B","C", & "D" IS COMPLETELY ARBITRARY)

, A/= EXCEPT[A/B , A/C] ,

THIS EXCLUDES A/B & A/C FROM ANY ACTION INVOLVING FILES WITH AN <MFID>=A. THE WORD "EXCEPT" IS NOT NECESSARY BUT MAY BE USED FOR SAKE OF CLARITY. A/= IS REFERRED TO AS THE PRIMARY FILE NAME IN THIS CASE.

, =/A AS =/B ,

THIS WOULD INDICATE THAT ALL FILES FOUND WITH <FID>=A SHOULD APPEAR ON THE DESTINATION WITH <FID>=B. HERE, =/A IS THE PRIMARY FILE NAME.

, =/= [=/A , B/C] ,

AGAIN THIS ILLUSTRATES USE OF AN "EXCEPT" LIST WHERE ALL FILES EXCEPT THE PARTICULAR FILE B/C AND THOSE WITH <FID>=A WILL BE ACTED UPON.

, =/A[B/A] AS =/C ,

HERE THE TWO MODIFIERS ARE COMBINED. THE NET RESULT WOULD BE THAT ALL FILES WITH <FID>=A, EXCEPT FOR THE FILE B/A, WOULD APPEAR ON A DESTINATION WITH AN <FID>=C.

(A) REMOVE CONTROL CARD MAKING USE OF THE "EXCEPT" FACILITY:

CC REMOVE A/=[A/B,A/C,A/D], =/B, =/C[B/C]; END

REMOVES ALL FILES WITH <MFID>=A EXCEPT A/B, A/C, & A/D, ALL FILES WITH <FID>=B AND ALL FILES WITH <FID>=C EXCEPT B/C.

(B) LIBRARY MAINTENANCE CONTROL CARDS USING MODIFIERS:

CC LOAD FROM TAPE1 =/A [A/A] AS =/B , =/B AS =/C; END

TRANSFER ALL THE FILES WITH <FID>=A EXCEPT A/A, CHANGING THEIR <FID>"S TO B, AND ALL FILES WITH <FID>=B CHANGING THEIR <FID>"S TO C FROM THE INPUT SOURCE TAPE NAMED "TAPE1" TO DISK.

CC COPY =/= [A/=] FROM DISK, =/B[C/B] AS =/C FROM T1 TO T2;END

TRANSFER ALL FILES EXCEPT THOSE WITH <MFID>=A FROM THE DISK, AND ALL FILES WITH <FID>=B, EXCEPT C/B, CHANGING THEIR <FID>"S TO C FROM A TAPE NAMED "T1" TO A TAPE NAMED "T2".

NOTES:

(A) EACH PRIMARY NAME WITHIN A NAME LIST MAY BE MODIFIED BY EITHER AN "AS" CLAUSE OR AN "EXCEPT" LIST OR BOTH.

(B) DUE TO THE ADDED COMPLEXITY OF THE RESULTING NAME LISTS USERS MAY GET "NOT ON DISK" MESSAGES ABOUT FILES WHICH ACTUALLY ARE ON DISK AND/OR "NOT ON TAPE" MESSAGES ABOUT FILES WHICH ARE ACTUALLY ON THE TAPE. SUCH MESSAGES WILL OCCUR IF, IN THE CASE OF A HASHED DISK SEARCH, THE "EXCEPT" LIST EXHAUSTS THE PRIMARY NAME, OR IN ANY OTHER TYPE OF ACCESSING TECHNIQUE, IF THE SAME FILE HAS BEEN ENCOUNTERED WITHIN A PARTICULAR NAME LIST MORE THAN ONCE.

(C) OPERATORS SHOULD TAKE NOTE THAT THE NAME SPOUTED BY THE DISK PARITY MESSAGE MAY NOT BE THE FILE WITH THE PARITY BECAUSE OF A POSSIBLE "AS" CLAUSE IN THE CONTROL CARD WHICH EVENTUALLY RESULTS IN THE PARITY.

PRECEDING THE NAME LIST FOR EACH SOURCE A LIST OF OPTIONS MAY BE SPECIFIED. THE OPTIONS AVAILABLE AND THE ORDER IN WHICH THEY MUST APPEAR IF PRESENT FOLLOWS:

1) "LATEST" = (APPLICABLE ONLY TO TRANSFERS GOING FROM A TAPE SOURCE TO DISK DESTINATION) TRANSFER THE FILE ONLY IF THE VERSION ON TAPE IS MORE RECENT.

2) "EXPIRED" = (APPLICABLE ONLY TO DISK SOURCE) TRANSFER THE FILE ONLY IF IT HAS EXPIRED.

3) "ACCESSD" = (APPLICABLE ONLY TO DISK SOURCE) TRANSFER THE FILE ONLY IF IT HAS BEEN MARKED AS ACCESSED.

4) "ADD" = (APPLICABLE ONLY TO TRANSFERS GOING FROM A TAPE SOURCE TO DISK DESTINATION) TRANSFER THE FILE ONLY IF IT IS NOT ALREADY ON DISK.

5) "NOHASH" = (APPLICABLE ONLY TO DISK SOURCE) SPECIFIES THAT NO HASHING TECHNIQUE SHOULD BE APPLIED WHEN FINDING FILES FOR THIS SOURCE. THIS OPTION SHOULD ONLY BE USED WHEN A GOOD PERCENTAGE OF THE DIRECTORY IS EXPECTED TO BE TRANSFERED.

6) "UNLOAD" = (APPLICABLE ONLY TO DISK SOURCE) REMOVES DISK FILES AFTER COMPLETION OF TRANSFER.

NOTE:

(A) IF AN OPTION IS SPECIFIED, BUT DOES NOT APPLY TO THE PARTICULAR SITUATION, IT WILL BE IGNORED.

(B) SPECIFICATION OF THE "B6500" OPTION IS NO LONGER NECESSARY. IT IS PRE-DETERMINED FOR EACH SOURCE AS IT IS MOUNTED.

EXAMPLE OF "COPY" CONTROL CARD USING OPTIONS:

```
CC COPY ACCESSD NOHASH UNLOAD =/A[B/A] FROM DISK,
=/B FROM TAPE1 TO TAPE2; END
```

OF THE TWO SOURCES ONLY THE DISK SOURCE HAS OPTIONS SPECIFIED. ALL FILES WITH <FID>=A, EXCEPT B/A WILL BE LOCATED ON DISK USING A SERIAL SEARCH RATHER THAN A HASHING TECHNIQUE. THIS LIST WILL BE FURTHER REFINED BY SELECTING ONLY THOSE FILES WHICH HAVE THE ACCESSD BIT ON IN THEIR HEADERS. AS EACH OF THESE FILES IS TRANSFERRED THEY WILL BE REMOVED FROM DISK. FINALLY ALL FILES WITH <FID>=B WILL BE TRANSFERRED FROM A TAPE NAMED "TAPE1" JOINING THE FILES FROM DISK ON A TAPE NAMED "TAPE2".

SO FAR WE HAVE BEEN DISCUSSING THOSE FEATURES APPLICABLE TO THE NAME LISTS PERTAINING TO PARTICULAR SOURCES OR TO OPTIONS PERTAINING TO A PARTICULAR SOURCE. NOW WE WILL EXAMINE THE COMBINATION OF THE SOURCES ALLOWED IN GREATER DETAIL.

SPECIFIED DESTINATION		SOURCES
DISK	CAN HAVE	ONE OR MORE TAPE SOURCES
TAPES	CAN HAVE	ONE OR MORE TAPE SOURCES AND/OR ONE OR MORE DISK SOURCES

NOTE:

ONE MAY SPECIFY MORE THAN ONE DISK SOURCE, BUT CARE SHOULD BE TAKEN TO AVOID "IN USE" CONFLICTS OCCURRING WHEN THE SAME NAME APPEARS IN MORE THAN ONE SOURCES NAME LIST. A "<MIX>IF" KEYIN WOULD BE NECESSARY TO CLEAR THESE CONFLICTS.

THE FOLLOWING IS AN EXAMPLE OF A "COPY" CONTROL CARD INVOLVING MORE THAN ONE SOURCE.

```
CC COPY =/= FROM TAPE1, =/A[B/A] AS =/B, C/= FROM TAPE2,
   EXPIRED B/= FROM DISK, C/= AS D/= FROM TAPE3 TO TAPE4;END
```

NOTE:

IF A FILE EXISTS ON MORE THAN ONE SOURCE THE ONE ENCOUNTERED FIRST WILL BE SELECTED. JUDICIAL USE OF THE "EXCEPT" AND "AS" FACILITIES ALONG WITH CORRECT PLACEMENT OF SOURCES SHOULD MAKE SELECTION OF DESIRED FILES FROM EACH SOURCE POSSIBLE, EVEN IF A FILE WITH THE SAME NAME AS THE ONE DESIRED EXISTS ON EVERY SOURCE.

EXAMPLE:

SUPPOSE ONE WANTED ALL THE FILES ON A TAPE NAMED "TAPE1", AND ALL THE FILES ON DISK WITH <FID>=D TRANSFERED TO A TAPE NAMED "TAPE2". IN PARTICULAR, A/B WAS DESIRED FROM "TAPE1" AND C/D WAS DESIRED FROM DISK WHERE A/B AND C/D EXIST ON BOTH SOURCES, THE FOLLOWING CONTROL CARD WOULD ACCOMPLISH THE DESIRED RESULT:

CC COPY =/=[C/D] FROM TAPE1, =/D FROM DISK TO TAPE2;END

IT IS OBVIOUS FROM THE EXAMPLES THAT NAME LISTS PRECEDE SOURCES IN THE "COPY" CONTROL CARD WHEREAS THEY FOLLOWED SOURCES IN OLD LIBRARY MAINTENANCE CARDS, BUT EQUIVALENCE HAS BEEN MAINTAINED. ALL OLD CONTROL CARDS ARE COMPATABLE WITH THE NEW LIBRARY MAINTENANCE.

EXAMPLE OF THREE EQUIVALENT STATEMENTS:

CC ADD FROM TAPE1 =/=; END

CC COPY ADD =/= FROM TAPE1 TO DISK; END

CC COPY ADD =/= FROM TAPE1; END

NOTE:

(A) IF A DISK SOURCE IS THE LAST SOURCE, IT NEED NOT BE SPECIFIED, ALSO, IF DESTINATION IS DISK IT NEED NOT BE EXPLICITLY STATED.

(B) TYPE OF DESTINATION (IF DISK) MAY BE SPECIFIED. TYPES AVAILABLE ARE "EU <NUMBER>", "FAST", AND "SLOW".

EXAMPLE: (TWO EQUIVALENT STATEMENTS)

```
CC LOAD TO EU 1 FROM TAPE1 =/D [ B/D ], C/= AS B/=; END

CC COPY =/B [B/D] , C/= AS B/= FROM TAPE1 TO EU 1; END
```

THE USER MAY SPECIFY THE MAXIMUM NUMBER OF FILES PER OUTPUT UNIT SIMPLY BY PUTTING THIS NUMBER DIRECTLY AFTER THE WORD "COPY". THIS NUMBER IS USED TO DETERMINE WHEN TO INITIATE ANOTHER VERSION OF LIBMAIN/DISK. THE ORIGINATING LIBMAIN/DISK WILL KEEP FILLING A LIST OF FILES UNTIL THE MAXIMUM IS EXCEEDED. AT WHICH TIME IT WILL DETERMINE IF IT CAN FORK ANOTHER LIBMAIN/DISK. IF IT IS WORKING ON A DISK SOURCE OR HAPPENS TO BE BETWEEN TWO TAPE SOURCES THE NEW LIBMAIN/DISK WILL BE FORKED. HOWEVER, IF IT IS STILL PROCESSING FILES ON THE SAME TAPE UNIT IT WOULD MEAN THAT, UPON FORKING ANOTHER JOB, TWO LIBMAIN/DISK"S WOULD BE COMPETING FOR THE SAME INPUT UNIT. IF THE LATTER OCCURS, AN ATTEMPT WILL BE MADE TO CUT BACK THE LIST BEING PROCESSED TO THE POINT WHERE THE LAST CHANGE OF INPUT SOURCES TOOK PLACE. IF THIS IS IMPOSSIBLE THE LIST MUST BE EXTENDED OVER THE MAXIMUM SPECIFIED.

EXAMPLE OF THE USE OF A MAXIMUM SPECIFICATION:

```
CC COPY 18 NOHASH =/B, =/C AS =/D FROM DISK, =/=[A/D]
   FROM TAPE1 TO TAPE2; END
```

IN THIS CASE, EACH REEL OF "TAPE2" WOULD CONTAIN 18 OR LESS FILES, UNLESS AT THE TIME 18 IS EXCEEDED THE CURRENT LIBMAIN/DISK IS WITHIN THE NAME LIST OF "TAPE1". IF THIS WAS THE CASE THE CURRENT REEL OF "TAPE2" WOULD CONTAIN ALL THE FILES FROM "TAPE1" (THAT WERE DESIRED) EVEN IF THE NUMBER EXCEEDED 18.

TECHNICAL DESCRIPTION

THE NEW LIBRARY MAINTENANCE CONSISTS OF FOUR MAIN PROCEDURES. CCLIB (OF CONTROLCARD) SCANS THE CONTROL CARD INPUT, STORING THE NAMES AND INFORMATION IN A LINKED LIST OF ESPDISK SEGMENTS, IN A FORMAT DISCUSSED LATER. LIBRARYCOPY USES THE INFORMATION PASSED BY CCLIB TO CREATE TABLES WHICH HAVE ALL THE INFORMATION NECESSARY TO PERFORM ACTUAL PHYSICAL TRANSFER. LIBRARYTRANSFER USES THE INFORMATION OBTAINED BY LIBRARYCOPY TO PERFORM THE DATA TRANSFER. BOTH ROUTINES USE LIBRARYHELP WHEN INFREQUENTLY USED CODE MUST BE DONE.

CCLIB

"CCLIB" WILL SCAN THE "COPY" CONTROL CARD ACCORDING TO THE FOLLOWING BNF DESCRIPTION... (THE SYMBOL "#" MEANS "OR")

```
<COPY FORM>            ::= COPY <MAXIMUM> <COPY SPECIFICATION>

<MAXIMUM>              ::= <EMPTY> # <NUMBER>

<COPY SPECIFICATION>   ::= <FROM-TO FORM> # <TO FORM> # <FROM FORM>
                           # <FROM FORM> , <COPY SPECIFICATION>

<FROM-TO FORM>         ::= <OPTION-LIST> FROM <VOLUME1> TO <VOLUME2>

<TO FORM>              ::= <OPTION-LIST> TO <VOLUME2>

<FROM FORM>            ::= <OPTION-LIST> FROM <VOLUME1>

<OPTION-LIST>          ::= <OPTIONS> <LIST>

<VOLUME1>              ::= DISK # <LIBRARY TAPE NAME>

<VOLUME2>              ::= DISK # FAST # SLOW # EU <NUMBER> #
                           <LIBRARY TAPE NAME>
```

```
<OPTIONS>           ::= <LATEST CLAUSE> <EXPIRED CLAUSE>
                        <ACCESSD CLAUSE> <ADD CLAUSE>
                        <NOHASH CLAUSE> <UNLOAD CLAUSE>

<LATEST CLAUSE>     ::= <EMPTY> # LATEST

<EXPIRED CLAUSE>    ::= <EMPTY> # EXPIRED

<ACCESSD CLAUSE>    ::= <EMPTY> # ACCESSD

<ADD CLAUSE>        ::= <EMPTY> # ADD

<NOHASH CLAUSE>     ::= <EMPTY> # NOHASH

<UNLOAD CLAUSE>     ::= <EMPTY> # UNLOAD

<LIST>              ::= <FILE SPECIFIER> # <FILE SPECIFIER> ,
                        <LIST>

<FILE SPECIFIER>    ::= <NAME PART> <EXCEPT PART> <AS PART>

<NAME PART>         ::= <MULTIFILE ID>/<FILE ID> # */<FILE ID>
                        <MULTIFILE ID>/* # */*

<EXCEPT PART>       ::= <EMPTY> # [ <EXCEPT LIST> ] #
                        EXCEPT [ <EXCEPT LIST> ]

<EXCEPT LIST>       ::= <NAME PART> # <NAME PART> , <EXCEPT LIST>

<AS PART>           ::= <EMPTY> # AS <NAME PART>

<NUMBER>            ::= <INTEGER>
```

NOTE:

IF "UNLOAD" AND/OR "NOHASH" ARE SPECIFIED, NO "AS" MODIFIER WILL BE ALLOWED WITHIN THE NAME LIST FOR THAT PARTICULAR SOURCE.

ESPDISK FORMAT:

THE SECOND WORD OF THE FIRST ESPDISK SEGMENT HOLDS THE USER SPECIFIED DESTINATION. FROM THEN ON UNTIL AN END OF LIST (@14) INDICATION THE WORDS ARE GROUPED IN PAIRS. THE FOUR TYPES ARE:

1) BEGINNING OF SOURCE PAIR
   WORD[1] = 0
   WORD[2] = .[3:6] OPTIONS FOR THIS SOURCE
   .[3:1] "ACCESSD"
   .[4:1] "EXPIRED"
   .[5:1] "LATEST"
   .[6:1] "ADD"
   .[7:1] "NOHASH"
   .[8:1] "UNLOAD"
   .[FF] COUNT OF "EXCEPT" OR "AS" NAMES
   .[CF] COUNT OF PRIMARY NAMES

2) NAME PAIR
   WORD[1] = MULTIFILE ID
   .[4:1] ASSOCIATED WITH THE "AS" CLAUSE
   .[5:1] ASSOCIATED WITH THE "EXCEPT" LIST
   WORD[2] = FILE ID
   (SAME AS WORD[1])

3) END OF SOURCE PAIR
   WORD[1] = @14
   WORD[2] = USER SPECIFIED SOURCE NAME

4) LINK PAIR
   WORD[1] = 0
   WORD[2] = ADDRESS OF NEXT ESPDISK SEGMENT

DESCRIPTION OF VARIABLES:

```
FROMHLD          .[2:1]  SOURCE DISK
                 IF [2:1]=1 THEN .[3:45]=0
                            ELSE .[3:45] IS TAPE NAME

TOHLD            .[2:1]  DESTINATION DISK
                 IF [2:1]=1 THEN .[40:1] SPECIFIES FAST DISK
                                 .[41:1]      "     SLOW DISK
                                 .[42:6]      "     EU # + 1
                            ELSE .[3:45] IS TAPE NAME

REMEMBER         .[3:15] ADDRESS OF FIRST ESPDISK SEGMENT FOR
                         PLACEMENT OF TOHLD
                 .[FF]   OFFSET INTO [CF]
                 .[CF]   ADDRESS OF ESPDISK SEGMENT INTO WHICH
                         OPTION WORD WILL BE PLACED FOR EACH SOURCE

NAMECNT          .[17:1] INDICATES <AS PART> NOT ALLOWED
                 .[FF]   NUMBER OF "EXCEPT" AND "AS" NAMES
                         ASSOCIATED WITH THIS INPUT SOURCE
                 .[CF]   NUMBER OF PRIMARY NAMES ASSOCIATED
                         WITH THIS INPUT SOURCE

XLST             ARRAY USED FOR HOLDING A SCANNED "EXCEPT" LIST
                 FOR BOTH "COPY" AND "REMOVE"

CMM[19]          (COMMON)
                 .[2:1]  SPECIFIES THIS IS ORIGINATING LIBMAIN/DISK
                 .[3:6]  UNITNO FOR PACKETS
                 .[9:9]  USER SPECIFIED MAXIMUM NUMBER OF FILES
                         PER OUTPUT UNIT
                 .[CF]   ADDRESS OF FIRST ESPDISK SEGMENT
```

"CCLIB" WILL SCAN "EXCEPT" LISTS ASSOCIATED WITH NAMES IN A "REMOVE" CONTROL CARD PASSING THIS LIST TO THE PROCEDURE "SEEKNAM", WHICH IN TURN USES THE LIST WHEN DETERMINING WHETHER OR NOT TO RETURN A PARTICULAR NAME TO "CCLIB" FOR REMOVAL.

LIBRARYCOPY

IT IS THE FUNCTION OF LIBRARYCOPY TO MANIPULATE THE DATA GIVEN IT BY CCLIB CREATING A FILE NAME TABLE (FA) AND FILE TRANSFER INFORMATION (FAINFO) WHICH IT WILL PASS TO LIBRARYTRANSFER FOR FINAL PROCESSING AND ACTUAL PHYSICAL TRANSFER.

FOR EACH SOURCE ENCOUNTERED IN THE ESPDISK LIST,

LIBRARYCOPY WILL:

A) BREAK THE FILE NAMES AND INFORMATION DOWN FROM THE FORM GIVEN BY CCLIB TO ONE IT CAN WORK WITH READILY,

AND:

B) RESOLVE WHETHER OR NOT ANY PARTICULAR FILE ON THE INPUT SOURCE SHOULD BE CONSIDERED A CANDIDATE FOR TRANSFER.

TO DO THIS IS ESSENTIALLY A SIX STAGE PROCESS...

1) DETERMINE IF A NAME PAIR FROM THE SOURCE HAS BEEN ASKED FOR BY THE CONTROL CARD,

2) CHECK TO SEE THAT THE NAME YOU ARE INVESTIGATING DOES NOT EXIST IN AN "EXCEPT" LIST,

3) IF DISK, MAKE SURE THE FILE IS ACCESSABLE,

4) CHANGE THE NAME OF THE FILE IF SPECIFIED BY AN "AS" PART,

5) CHECK THAT THIS NAME PAIR HAS NOT ALREADY BEEN PROCESSED,

AND:

6) INSERT THE NAME INTO THE FILE TABLE (FA), INSERT TRANSFER INFORMATION FOR THIS NAME INTO THE INFORMATION TABLE (FAINFO).

AT STAGE 6, IT MUST BE DETERMINED AS TO WHETHER THE USER SPECIFIED MAXIMUM NUMBER OF FILES PER OUTPUT UNIT HAS BEEN EXCEEDED. THE DEFAULT MAXIMUM IS 511 FILES. IF THE NUMBER HAS BEEN EXCEEDED, LIBRARYCOPY WILL ATTEMPT TO FORK ANOTHER LIBMAIN/DISK TO HANDLE THE CURRENT FILE TABLE (FA). IN THIS SENSE THE MAXIMUM SPECIFIED BY THE USER MUST BE CONSIDERED APPROXIMATE FOR WE DO NOT WISH FILES FROM THE SAME INPUT TAPE SOURCE TO EXIST ON TWO SEPATATE "FA"S (ONE LIBMAIN/DISK WOULD HAVE TO WAIT FOR THE OTHER TO FINISH WITH THE TAPE). IF THE MAXIMUM IS EXCEEDED BETWEEN TWO TAPES, BETWEEN DISK AND TAPE, OR WHILE PROCESSING A DISK SOURCE THE CURRENT "FA" WOULD BE PASSED (IN USER DISK) TO THE FORKED LIBMAIN/DISK WITH A LIST OF FILES EQUALING THE MAXIMUM SPECIFIED. HOWEVER, IF IT IS STILL PROCESSING WITHIN A TAPE SOURCE THE "FA" WILL BE CUT BACK TO THE LAST CHANGE OF SOURCES IN THAT "FA", AND IF THERE HAS BEEN NO CHANGE, IT MUST KEEP INSERTING WITHIN THE CURRENT "FA" UNTIL A CHANGE OCCURS. IN THE LATTER CASE, THE MAXIMUM MUST BE IGNORED. WHEN THE FORKING HAS BEEN COMPLETED THE PROCESSING WILL CONTINUE WITH A NEW "FA" AND A LINKED LIST OF THESE "FA"S WILL BE CREATED BY THE ORIGINATING LIBMAIN/DISK TO HANDLE CONFLICT RESOLUTION DURING STAGE 5.

AT STAGE 1, THE FIRST STEP IS TO OBTAIN A SOURCE FILE. THERE ARE TWO METHODS USED. THE FIRST IS STRICTLY FOR DISK, MAKING USE OF THE PROCEDURE "SEEKNAM" TO LOCATE CANDIDATE FILES. THE SECOND IS THE METHOD FOR TAPE IN WHICH THE DIRECTORY IS READ AND FILES ARE EXAMINED ONE AT A TIME FROM THAT DIRECTORY. BY USING THE OPTION "NOHASH", ONE SPECIFIES THAT HE WISHES TO USE THE SERIAL METHOD DESCRIBED ABOVE FOR TAPE, ON THE DISK; TREATING THE DISK DIRECTORY LIKE A LARGE TAPE DIRECTORY. SUCH A METHOD SHOULD BE USED ONLY WHEN IT IS KNOWN THAT A SUBSTANTIAL PORTION OF THE DISK DIRECTORY WILL BE INVOLVED IN THE TRANSFER. THIS METHOD ESSENTIALLY ELIMINATES THE POSSIBILITY OF READING AND WRITING THE SAME SEGMENT TWICE.

THE FPB ENTRIES ARE HANDLED DIFFERENTLY DEPENDING ON WHETHER THE LIBMAIN/DISK IN QUESTION IS THE ORIGINATING VERSION OR AN OFFSPRING. THE ORIGINATING LIBRARYCOPY STARTS TIMING ON EACH SOURCE AS IT IS ENCOUNTERED. IF THE FILES ASSOCIATED WITH A PARTICULAR SOURCE ARE FORKED TO ANOTHER LIBMAIN/DISK A STOP TIMING IS DONE BUT THE TAPES ARE LEFT IN USE. WHEN THE FORKED JOB BEGINS IT STARTS TIMING ON THE TAPES LEFT IN USE AND CONTINUES FROM THAT POINT. A NON-FORKED LIBMAIN/DISK WILL EXPAND THE FPB ITSELF WHEN NECESSARY, WHEREAS A FORKED OFFSPRING WILL MAKE USE OF THE SELF EXPANDING CAPABILITIES OF "STARTIMING".

DESCRIPTION OF VARIABLES:

```
   COMMON.[2:1]    INDICATES ORIGINATING LIBMAIN/DISK

IF ORIGINATOR...

         .[3:6]    UNITNO FOR PACKETS
         .[9:9]    MAXIMUM NUMBER OF FILES PER OUTPUT UNIT
         .[CF]     FIRST ESPDISK SEGMENT ADDRESS

IF NOT ORIGINATOR...

         .[3:45]   DISK ADDRESS OF "FA" AND "FAINFO" FOR THIS
                   FORKED LIBMAIN/DISK

   FA              FILE TABLE FOR NAME PAIRS
   FAINFO          TRANSFER INFORMATION FOR ASSOCIATED NAME PAIR
   FAIN            INDEX INTO "FA"
   FASZ            SIZE OF "FA"
   MFID            MULTI-FILE ID
   FID             FILE ID
   ASMFID          MULTI-FILE ID AFTER STAGE 4.
   ASFID           FILE ID AFTER STAGE 4.
   FPBPTR          CURRENT INDEX OF FPB ENTRY
   DESTIN          USER SPECIFIED DESTINATION

   TOGS.[3:1] (ACCESSD)    SPECIFIES CHECK ACCESSD BIT
       .[4:1] (EXPIRED)    CHECK FOR FILE BEING EXPIRED
       .[7:1] (NOHASH)     USE DISK SERIALLY
       .[18:1] (OK)        FILE HAS PASSED STAGE 3.
       .[19:1] (INXLST)    FILE OCCURRED IN "EXCEPT" LIST
       .[20:1] (WEIRDFORK) SOME ALTERATIONS TO "FA" ARE
                           NECESSARY BEFORE FORKING
       .[23:1] (FORKED)    NOT ORIGINATING LIBRARYCOPY
       .[25:1] (SOURCEFILEFOUND) AT LEAST ONE FILE FROM THIS
                           SOURCE WAS USED

   CCA             HOLDS ESPDISK SEGMENTS FROM CCLIB

   NA,
   EA,
   POOL            USED IN PREPROCESSING CONTROL CARD INFO. FROM
                   CCLIB FOR USE BY LIBRARYCOPY

   MAX             USER SPECIFIED MAXIMUM NUMBER OF FILES PER
                   OUTPUT UNIT

   X               DIRECTORY OF CURRENT SOURCE (TAPE OR NOHASH)
```

```
                    **********************
                    *                    *
                    **  LIBRARYTRANSFER  **
                    *                    *
                    **********************

THIS PROCEDURE, GIVEN A FILE TABLE (FA) AND ITS ASSOCIATED
INFORMATION TABLE (FAINFO), HAS THE CAPABILITY OF TRANSFERING DATA
FROM DISK TO TAPE, TAPE TO DISK, OR TAPE TO TAPE.  IT ESSENTIALLY
ACTS ON EACH ENTRY FROM "FAINFO" ONE AT A TIME UNTIL THEY HAVE BEEN
EXHAUSTED, PERFORMING THE PHYSICAL TRANSFER OF DATA AS SPECIFIED
WITHIN THE ENTRY.  FIXUPS TO THE FPB OCCUR AT EACH CHANGE OF INPUT
UNIT AS DETECTED WITHIN THE "FAINFO" TABLE, HOWEVER, THE METHODS
DIFFER DEPENDING UPON WHETHER THE CURRENT JOB IS AN OFFSPRING OR THE
ORIGINATOR.  DESCRIPTION OF VARIABLES:

  DA              CURRENT "FAINFO" ENTRY
    .[CF]         IF DISK THEN 18
                  IF TAPE THEN UNIT NUMBER OF THE TAPE
    .[FF]         IF DISK THEN DISK ADDRESS OF FILE HEADER
                  IF TAPE THEN NUMBER OF THIS FILE ON TAPE
    .[5:1]        SPECIFIES LATEST VERSION WANTED
    .[6:1]        FILE TO BE ADDED
    .[8:1]        FILE TO BE UNLOADED

  IU              CURRENT INPUT UNIT
  OU              "     OUTPUT  "
  IREEL           CURRENT INPUT REEL NUMBER (IF TAPE)
  OREEL           "     OUTPUT  "    "         "
  FA              FILE TABLE OF NAME PAIRS TO BE TRANSFERRED
  FAINFO          TRANSFER INFO. FOR EACH NAME PAIR
  FPBPTR          CURRENT FPB ENTRY INDEX
  H               CURRENT FILE HEADER
  LAB             LAST INPUT LABEL READ
  LBL             OUTPUT LABEL TO BE WRITTEN

  TOGS.[21:1] (BHS)          INDICATES BAD HEADER
       [23:1] (FORKED)       NOT ORIGINATING LIBMAIN/DISK
       [26:1] (OE)           CURRENT <MFID>/<FID> HAS BEEN
                             OPENED EXCLUSIVE

       [28:1] (SOMECOPIED)   AT LEAST ONE FILE HAS BEEN
                             TRANSFERED
       [38:1] (COPYING)      NOTES STAGE OF PROCESSING FOR
                             USE BY ABORT
       [40:1] (SKIPFILE)     USED TO INDICATE ABRUPT EXIT TO
                             NEXT FILE SHOULD BE TAKEN
```

LIBRARYHELP

THIS ROUTINE PERFORMS INFREQUENTLY NEEDED TASKS FOR BOTH "LIBRARYCOPY" AND "LIBRARYTRANSFER", A DESCRIPTION OF THE CASES IT HANDLES FOLLOWS:

0: A) SETS UP FPB ENTRY FOR INPUT SOURCE TAPE
B) RETURNS DIRECTORY LEAVING TAPE POSITIONED AFTER ENDING LABEL OF DIRECTORY

1: RETURNS AFTER SECURING A NEW INPUT SOURCE TAPE

2: ABORT FROM LIBRARYTRANSFER

3: SETS UP FPB ENTRY FOR INPUT SOURCE DISK

4: ABORT FROM LIBRARYCOPY

5: INITIALIZATION OF LIBRARYTRANSFER INCLUDING LOCATION OF SPECIFIED OUTPUT UNIT

6: ERROR HANDLING OR REEL SWITCHING

7: WRITING DIRECTORY IF NOT REEL 1

8: EXTRA RECORDS DETECTED IN CURRENT FILE

9: BAD HEADER DETECTED ON SOURCE

10: INVALID RECORD SIZE ON LAST READ OR WRITE

11: KEYIN REEL SWITCH

NOTE:

ALL THREE OF THE AFORE MENTIONED ROUTINES USE THE SAME STACK. THIS IS POSSIBLE BECAUSE "LIBRARYCOPY" AND "LIBRARYTRANSFER" WILL NEVER BE ACTIVE AT THE SAME TIME FOR THE SAME MIX, I.E. "LIBRARYTRANSFER" IS NEVER INITIATED UNTIL "LIBRARYCOPY" HAS COMPLETED THE PRE-PROCESSING.

APPENDIX B - CHANGES TO "UPDATE/USERS"

CHANGES TO "UPDATE/USERS"

INTRODUCTION

DCMCP NOT APPLICABLE TO TIMESHARING

TO HELP ALLEVIATE POTENTIAL NO-MEM PROBLEMS, THE MCP KEYIN ROUTINES HAVE BEEN ALTERED TO ALLOW FOR THE USE OF THREE SEPERATE PROCEDURES. THESE PROCEDURES ARE CALLED "KEYIN0", "KEYIN1", AND "KEYIN2". THE "MIXMSG" AND "INFOMSG" TABLES HAVE ALSO BEEN COMBINED INTO A SINGLE TABLE CALLED "KEYINTABLE".

THEREFORE, INSTEAD OF SPECIFYING WHICH "INFO" MESSAGES AND WHICH "MIX" MESSAGES A REMOTE USER IS PERMITTED TO USE, THE SITE MUST NOW SPECIFY WHICH "KEYIN" MESSAGES ARE PERMITTED THROUGH THE USE OF THE MODIFIED "UPDATE/USERS" PROGRAM.

THE "UPDATE/USERS" PROGRAM WILL NOW RECOGNIZE THE FOLLOWING CODE NAMES WHICH SHOULD APPEAR IN COLUMNS 73-80 OF BINARY CARDS SPECIFYING THE OPTIONS:

"CCMASK"
"KEYIN0"
"KEYIN1"
"KEYIN2"

ALSO WITH THE ADDITION OF THE COPY CONTROL CARD SEVERAL CHANGES WERE MADE TO THE "CCMASK" BINARY CARD. THE CARD COLUMNS FOR THE "CCMASK" BINARY CARD SPECIFY WHICH CONTROL CARD OPTIONS A REMOTE USER IS PERMITTED TO USE. THE CARD COLUMNS FOR THE "KEYIN" AND "CCMASK" OPTIONS AND THEIR CORRESPONDING NAMES FOLLOW.

"KEYIN0" BINARY CARD

COLUMN	WORD	KEYINOMASK BIT	STANDARD MASK	
2	AX	1	NO	REPLY TO JOB
3	IL	2	NO	LABEL
4	UL	3	NO	LABEL
5	QT	4	NO	QUIT PBACKUP
6	OL	5	NO	LABEL
7	WY	6	YES	WHY
8	RY	7	NO	READY
9	DS	8	NO	DISCONTINUE
10	SD	9	NO	DS BUT SAVE PACKET
11	TF	10	YES	TYPE CORE FACTOR
12	SF	11	NO	SET CORE FACTOR
13	RM	12	NO	REMOVE DUP FILE (DCMCP)
14	DP	13	NO	DUMP CORE
15	DD	14	NO	} DEBUGGING STUFF
16	DB	15	NO	
17	PT	16	NO	PRINT TRACE
18	ST	17	NO	SUSPEND JOB TEMPORARILY
19	CM	18	NO	CHANGE MCP
20	SV	19	NO	SAVE UNIT
21	CL	20	NO	CLEAR UNIT
22	BK	21	NO	BREAK
23	OK	22	NO	OK
24	FM	23	NO	REPLY TO FORMS MSG
25	FR	24	NO	FINAL REEL
26	OF	25	NO	OPTIONAL FILE
27	TI	26	YES	TIME FOR RUNNING JOB
28	PR	27	NO	PRIORITY
29	RO	28	NO	RESET OPTION
30	SD	29	NO	SEEMS TO DUPLICATE BIT #8
31	IT	30	NO	INTERRUPT ONLINE/MAINT
32	WI	31	YES	WHAT INTRINSICS?
33	IF	32	NO	IGNORE FILE FOR LIBMAIN
34	RC	33	NO	REEL CHANGE, TOO MANY ERRORS

"KEYIN1" BINARY CARD

COLUMN	WORD	KEYIN1MASK BIT	STANDARD MASK	
2	BO	1	YES	BLACKOUT FOR LOGIN
3	LI	2	YES	LOGIN
4	LO	3	YES	LOGOUT
5	ZZ	4	YES	SENSE EOT
6	TO	5	YES	TYPE OPTIONS
7	HM	6	YES	HALT MESSAGES
8	HR	7	YES	HALT MIX MSGS
9	SS	8	YES	STN-STN MESSAGE
10	BS	9	NO	BACKUP SPO, SET TERMINAL AS
11	US	10	NO	UNDO A BS
12	SC	11	NO	WHO'S A SPO?
13	QV	12	NO	TIME TO CLEAR READ-READY INTRPTS
14	RR	13	NO	UN-DESIGNATE REMOTE SPO CAPABILITY
15	WA	14	NO	WHAT STNS ARE ASSIGNED TO THIS JOB?
16	CA	15	NO	AUXMGM STUFF
17	DT	16	NO	ENTER DATE
18	WD	17	YES	WHATS DATE?
19	TR	18	NO	ENTER TIME
20	WT	19	YES	WHATS TIME?
21	HM	20	YES	HALT MESSAGES
22	CC	21	YES	CONTROL CARD FOLLOWS
23	CL	22	YES	CLEAR- SEEMS TO DUPLICATE KEYIN0 BIT 20
24	PB	23	NO	PRINT A BACKUP FILE
25	RN	24	NO	START PSEUDO-RDRS, OR ASK HOW MANY
26	LD	25	NO	START LDRNTRL/DISK
27	RD	26	NO	REMOVE DECK
28	RP	27	NO	REMOVE PACKET
29	ED	28	NO	ELIMINATE DECK FROM PSEUDO-RDR
30	SI	29	NO	SET STATISTICS INTERVAL
31	LR	30	NO	LOG REMOTE
32	OT	31	YES	LOOK AT A WORD IN PGM'S STACK
33	IN	32	NO	INPUT TO A PGM'S STACK
34	FE	33	NO	MAKE COMMENT IN MAINT/LOG
35	OC	34	NO	MAKE COMMENT IN LOG
36	SQ	35	NO	DISK SQUASH
37	CS	36	NO	CREATE SEPTICTANK
38	HS	37	NO	HALT SEPTIC
39	WK	38	NO	WORKSET REQUESTS

"KEYIN2" BINARY CARD

COLUMN	WORD	KEYIN2MASK BIT	STANDARD MASK	
2	WU	1	NO	WHO'S ON? WHAT'S UP?
3	WP	2	NO	WHAT PGMS ARE ASGD TO THIS STATION?
4	WR	3	NO	WHAT REMOTE LOG?
5	MX	4	YES	MIX
6	TS	5	YES	TYPE SCHEDULE
7	PS	6	NO	CHANGE PRIORITY IN SCHEDULE
8	ES	7	NO	ELIMINATE SCHEDULED JOB
9	XS	8	NO	FORCE EXECUTION FROM SCHEDULE
10	LF	9	NO	LIST FILES
11	LC	10	NO	LIST FOR CREATOR
12	LS	11	NO	LIST FOR SECURITY
13	EX	12	YES	[redacted] LIST EXPIRED FILES
14	PD	13	YES	PRINT DIRECTORY
15	SM	14	YES	START MAX MSGS (UNDO #m)
16	TO	15	YES	TYPE OPTIONS SETTINGS
17	PO	16	YES	PRINT A SPECIFIC OPTION
18	PG	17	NO	PURGE TAPE
19	AU	18	NO	AUXMGM
20	MS	19	NO	SET/RESET SYSTEM MONITOR
21	LN	20	NO	LOGOUT
22	CD	21	YES	PRINT PSEUDO DECKS
23	PP	22	NO	PRINT PACKETS
24	PC	23	NO	PACKET COUNT
25	CU	24	NO	CORE IN USE
26	SY	25	NO	WRITE OUT STATISTICS
27	SL	26	NO	" " "
28	RW	27	NO	REWIND TAPE
29	CI	28	NO	CHANGE INTRINSICS
30	CT	29	NO	CHANGE TIME FACTOR
31	XT	30	NO	EXTEND TIME FACTOR
32	TL	31	NO	TYPE LIMITS FOR JOB
33	XD	32	NO	MARK BAD DISK
34	MR	33	NO	MAKE RESCUE/DISK
35	MC	34	NO	MAKE PGM A COMPILER
36	RS	35	NO	READ BREAK FILE TO DISK
37	HD	36	NO	HOW MUCH DISK?
38	SA	37	NO	TYPE SEG # & ADDRESS

"CCMASK" BINARY CARD

COLUMN	WORD	CCMASK1 BIT	STANDARD MASK
22	UNLOCK	21	NO
23	USE	22	NO
24	LOCK	23	NO
25	FREE	24	NO
26	PUBLIC	25	NO
27	PACKET	26	NO
28	USER	27	NO
29	RUN	28	NO
30	COMPILE	29	YES
31	EXECUTE	30	YES
32	COPY	31	YES
33	DUMP	32	NO
34	UNLOAD	33	NO
35	ADD	34	NO
36	LOAD	35	NO
37	REMOVE	36	NO
38	CHANGE	37	NO
39	UNIT	38	NO
40	END	39	YES
41	WAIT	40	NO
42	DATA	41	NO
43	LABEL	42	YES
44	SET	43	NO
45	RESET	44	NO
46		45	NO
47		46	NO
48	FILE	47	NO

"CCMASK" BINARY CARD

COLUMN	WORD	CCMASK2 BIT	STANDARD MASK
50	ACCESSD	1	NO
51	PROCESS	2	YES
52	IO	3	NO
53	PRIORITY	4	NO
54	COMMON	5	YES
55	CORE	6	NO
56	STACK	7	NO
57	SAVE	8	NO
58		9	NO
59		10	NO
60		11	NO
61	ALGOL	12	YES
62	XALGOL	13	YES
63	FORTRAN	14	NO
64	TSPOL	15	NO
65	BASIC	16	NO
66	COBOL68	17	YES
67	WITH	18	NO
68	COBOL	19	YES
69	LIBRARY	20	YES
70	SYNTAX	21	NO
71	FROM	22	NO
72	TO	23	NO