.REM @

IDENTIFICATION

PRODUCT CODE: AC-F131C-MC

PRODUCT NAME: CZRLLCO RL01/02 DRIVE COMPATABILITY

DATE CREATED: 5-JAN-79

REVISED: 4-FEB-82

MAINTAINER: DIAGNOSTIC ENGINEERING - COLORADO

AUTHORS: D. DEKNIS, C. CAMPBELL

TABLE OF CONTENTS

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

1.1.1 STRUCTURE OF PROGRAM

THIS DIAGNOSTIC IS COMPATIBLE WITH BOTH XXDP+ AND ACT. IT CAN BE RUN STANDALONE UNDER XXDP+, AND CAN BE CHAINED UNDER XXDP+, ACT AND APT IN ACT MODE (SEE 2.2 "CHAIN MODE OPERATION" FOR DETAILS OF CHAINING PROCEDURE). IT IS A SINGLE PROGRAM FROM THE STANDPOINT OF THE DIAGNOSTIC USER, WHICH AT RUN TIME IS APPENDED TO A COMMON FRONT-END PIECE OF SUPERVISOR SOFTWARE THROUGH WHICH THE DIAGNOSTIC PROGRAM INTERFACES TO THE ENVIRONMENT AS IT EXECUTES.

WHEN THIS DIAGNOSTIC IS STARTED, CONTROL GOES FIRST TO THE SUPERVISOR PORTION, WHICH WILL ASK CERTAIN "HARD CORE" QUESTIONS ABOUT THE ENVIRONMENT. THEN IT WILL ENTER COMMAND MODE, INDICATED BY A PROMPT CHARACTER (DR>). AT COMMAND MODE THE OPERATOR MAY ENTER ANY OF SEVERAL COMMANDS AS DESCRIBED IN 2.0 "OPERATING INSTRUCTIONS".

THE DIAGNOSTIC PROGRAM IS LOADED IN THE LOWER 8K OF MEMORY. THE DIAGNOSTIC SUPERVISOR CODING OCCCUPIES 6.25K OF THE UPPER PART OF MEMORY JUST BELOW THE XXDP+ MONITOR WHICH RESIDES IN THE UPPERMOST 1.5K OF MEMORY SPACE.

1.1.2 DIAGNOSTIC INFORMATION

THE RL01 DRIVE COMPATABILITY TEST IS A PDP-11 (LSI-11) BASED PROGRAM THAT WILL TEST INTERCHANGEABILITY OF CARTRIDGES BETWEEN DRIVES. THE TEST PERFORMS WRITES, READS, OVERWRITES, ADJACENT CYLINDER WRITES TO PROVE COMPATABILITY. SINCE THE PROGRAM RELIES ON MANUAL INTERVENTION, A TOTAL TEST TIME IS NOT APPLICABLE. HOWEVER, TO TEST TWO DRIVES REQUIRES A MINIMUM OF THREE PASS. EACH PASS REQUIRES APPROXIMATELY 70 SECONDS.

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE REQUIREMENTS

* PDP-11/LSI-11 PROCESSOR WITH 16K OR MORE OF MEMORY
* CONSOLE DEVICE (LA30,LA36,VT50,ETC.)

* 1 OR 2 RL11/RLV11 CONTROLLER(S) WITH:

  1 - 8 RL01 DRIVES WITH RL01K CARTRIDGES CONTAINING A 'BAD SECTOR FILE'
  1 - 8 RL02 DRIVES WITH RL02K CARTRIDGES CONTAINING A 'BAD SECTOR FILE'

* LINE PRINTER (OPTIONAL)

1.2.2 SOFTWARE REQUIREMENTS

CZRLLC0 RL01/02 DRIVE COMPATABILITY
(FORMERLY CZRLFB)

1.3 RELATED DOCUMENTS AND STANDARDS

RL01 DISK SUBSYSTEM USER'S GUIDE (EK-RL01-UG-002)
XXDP+/SUPERVISOR USER'S MANUAL

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

THE RL01/02 SUBSYSTEM SHOULD HAVE SUCCESSFULLY RUN THE FOLLOWING PROGRAMS:

CVRLAB0	RLV11 RL01/02 DISKLESS TEST (RLV11 ONLY)
CZRLGC0	RL11/RLV11 RL01/02 CONTROLLER TEST (PART 1)
CZRLHC0	RL11/RLV11 RL01/02 CONTROLLER TEST (PART 2)
CZRLID0	RL01/02 DRIVE TEST (PART 1)
CZRLJB0	RL01/02 DRIVE TEST (PART 2)
CZRLKB0	RL11/RLV11 RL01/02 PERFORMANCE EXERCISER
CZRLNA0	RL01/02 DRIVE TEST (PART 3)

1.5 ASSUMPTIONS

THE HARDWARE OTHER THAN THE RL01/02 SUBSYSTEM IS ASSUMED TO WORK PROPERLY. FALSE ERRORS MAY BE REPORTED IF THE PROCESSOR, ETC., DO NOT FUNCTION PROPERLY.

2.0 OPERATING INSTRUCTIONS

2.1 HOW TO RUN THIS DIAGNOSTIC

2.1.1 THE FIVE STEPS OF EXECUTION

THIS DIAGNOSTIC PROGRAM SHOULD BE LOADED AND STARTED USING NORMAL XXDP+ PROCEDURES. START THE EXECUTION OF THE XXDP+ MONITOR BY USING THE APPROPRIATE BOOTSTRAP PROGRAM. THE MONITOR WILL PRINT A MESSAGE IDENTIFYING ITSELF AND REQUESTING THAT THE CURRENT DATE BE ENTERED. AN EXAMPLE OF THIS MESSAGE IS GIVEN BELOW FOR THE XXDP+ MONITOR:

```
CHMDKA0 XXDP+ MONITOR
BOOTED VIA UNIT 0
ENTER DATE (DD-MMM-YY):
```

AFTER THE DATE HAS BEEN ACCEPTED BY THE MONITOR THE RESTART ADDRESS OF THE MONITOR IS PRINTED. THEN THE FOLLOWING TWO QUESTIONS ARE ASKED:

```
50 HZ ? N
LSI ?  N
```

THE DEFAULTS ARE BOTH "NO". TYPE "R" AND THE PROGRAM NAME TO RUN THE PROGRAM. DO NOT TYPE THE EXTENSION.

WHEN THIS DIAGNOSTIC IS STARTED THE FOLLOWING 5 STEPS WILL OCCUR:

```
**********
* STEP 1 *
**********
```

THE DIAGNOSTIC WILL ISSUE THE PROMPT "DR>". FROM THIS POINT UNTIL THE TIME WHEN YOU RESTART XXDP+, YOU WILL BE TALKING TO THE DIAGNOSTIC, NOT XXDP+. WE WILL REFER TO THE PRESENCE OF THIS PROMPT AS BEING IN DIAGNOSTIC COMMAND MODE, AS OPPOSED TO XXDP+ COMMAND MODE.

AT THIS POINT YOU WILL ENTER A "START" COMMAND. THIS IS NOT THE SAME AS THE XXDP+ "START" COMMAND, WHICH YOU ALREADY ISSUED IN RESPONSE TO THE XXDP+ DOT PROMPT. THIS "START" COMMAND CAN TAKE A NUMBER OF SWITCHES AND FLAGS (ALL OPTIONAL) AND THE DETAILS OF THESE ARE SET FORTH IN 2.3 "DETAILS OF COMMANDS AND SYNTAX". HOWEVER, IN ORDER TO USE THE PROGRAM, ALL YOU NEED TO SAY IS SOMETHING LIKE THIS:

```
STA/PASS:1/FLAGS:HOE
```

THINGS TO NOTE HERE:

1. ONLY THE FIRST THREE CHARACTERS OF THIS OR ANY COMMAND AT THE "DR>" LEVEL NEED TO BE TYPED.

2. THE "PASS" SWITCH SPECIFIES HOW MANY PASSES YOU DESIRE. A PASS CONSISTS OF RUNNING THE FULL DIAGNOSTIC AGAINST ALL UNITS BEING TESTED (THIS WILL BE EXPLAINED SHORTLY). ONE PASS IS SPECIFIED IN THE ABOVE EXAMPLE.

3. THE "FLAGS" SWITCH MAY SPECIFY ANY OF A NUMBER OF FLAGS, BUT THE MAIN USEFUL ONES ARE:

PNT	PRINT NUMBER OF TEST BEING EXECUTED
LOE	LOOP ON ERROR
HOE	HALT ON ERROR
IER	INHIBIT ERROR PRINTOUT

THE HOE FLAG IS SPECIFIED IN THE ABOVE EXAMPLE (WE'LL SEE WHY SHORTLY).

****
* STEP 2 *
****

WHEN YOU HAVE TYPED IN A "START" COMMAND, THE DIAGNOSTIC WILL COME BACK WITH THE QUESTION "# UNITS?" TO WHICH YOU SHOULD RESPOND BY TYPING IN THE NUMBER OF DEVICES YOU WISH TO TEST.

A WORD OF WARNING HERE: THE NUMBER OF UNITS DEPENDS ON THE TARGET DEVICE OF THE DIAGNOSTIC. FOR EXAMPLE, IF THE DIAGNOSTIC IS DIRECTED AT A DISK DRIVE, THEN THE NUMBER OF UNITS WOULD BE THE NUMBER OF DRIVES TO BE TESTED. WHEREAS IF THE DIAGNOSTIC WAS DIRECTED AT THE DISK CONTROLLER, THEN THE NUMBER OF UNITS WOULD BE THE NUMBER OF CONTROLLERS. THE TARGET DEVICE OF A DIAGNOSTIC CAN ALWAYS BE DETERMINED BY INSPECTING THE "HEADER" STATEMENT NEAR THE BEGINNING OF THE SOURCE CODE. ONE OF THE OPERANDS OF THIS "HEADER" STATEMENT SHOULD BE THE DEVICE TYPE OF THE DIAGNOSTIC.

****
* STEP 3 *
****

WHEN YOU HAVE TYPED IN THE NUMBER OF UNITS TO BE TESTED, THE DIAGNOSTIC WILL ASK YOU THE "HARDWARE QUESTIONS". THE ANSWERS TO THESE QUESTIONS ARE USED TO BUILD TABLES IN CORE, CALLED "HARDWARE P-TABLES". ONE HARDWARE P-TABLE WILL BE BUILT FOR EACH UNIT TO BE TESTED.

THERE ARE SEVERAL HARDWARE QUESTIONS AND THE ENTIRE SERIES WILL BE POSED N TIMES, WHERE N IS THE NUMBER OF UNITS.

THIS REPRESENTS A NEW PHILOSOPHY IN DIAGNOSTIC ENGINEERING. DIAGNOSTICS IN THE FUTURE WILL NOT BE WRITTEN TO AUTOSIZE OR ASSUME STANDARD ADDRESSES: INSTEAD, THEY WILL ASK THE OPERATOR FOR ALL THE INFORMATION THEY NEED TO TEST THE DEVICE.

****
* STEP 4 *
****

AFTER YOU HAVE ANSWERED ALL THE HARDWARE QUESTIONS (SEC 2.5) FOR ALL THE UNITS, YOU WILL BE ASKED "CHANGE SW?" IF YOU WANT TO BE ASKED THE SOFTWARE QUESTIONS THAT DETERMINE THE BEHAVIOR OF THIS PROGRAM, TYPE 'Y". IF YOU WANT TO TAKE ALL THE DEFAULTS TO THESE QUESTIONS, TYPE 'N". IF YOU TYPE 'Y" YOU WILL BE ASKED THE SOFTWARE QUESTIONS (SEC 2.6), AND THE ANSWERS WILL BE PUT INTO THE SOFTWARE P-TABLE IN THE PROGRAM. THE SERIES OF QUESTIONS WILL BE ASKED JUST ONCE, REGARDLESS OF THE NUMBER OF UNITS TO BE TESTED.

****
* STEP 5 *
****

AFTER YOU HAVE ANSWERED THE SOFTWARE QUESTIONS, THE DIAGNOSTIC WILL BEGIN TO EXECUTE THE HARDWARE TEST CODE. THERE ARE SEVERAL THINGS THAT CAN HAPPEN NEXT, DEPENDING ON WHETHER A HARDWARE ERROR IS ENCOUNTERED AND ALSO ON WHAT SWITCH VALUES YOU SELECTED ON THE START COMMAND. CONSIDER THE POSSIBILITIES:

1. IF NO ERROR IS ENCOUNTERED, THEN THE DIAGNOSTIC WILL SIMPLY EXECUTE THE DESIRED NUMBER OF PASSES AND RETURN TO COMMAND MODE (PROMPT DR>).

2. IF AN ERROR IS ENCOUNTERED, THEN ONE OF THREE THINGS HAPPENS, DEPENDING ON THE SETTINGS OF THE HOE AND LOE FLAGS.

   HOE SET: THE ERROR WILL BE REPORTED ON THE CONSOLE AND THE DIAGNOSTIC WILL RETURN TO COMMAND MODE.

   LOE SET: THE DIAGNOSTIC WILL LOOP ENDLESSLY ON THE BLOCK OF CODE THAT DETECTED THE ERROR.

   NEITHER HOE NOR LCE SET: THE ERROR WILL BE REPORTED ON THE CONSOLE AND NORMAL EXECUTION WILL RESUME AS IF NO ERROR HAD OCCURRED.

2.1.2 SAMPLE RUN-THROUGH

LET'S SEE HOW ALL THIS WORKS IN A REAL SITUATION. RECALL THAT WE ENTERED THE COMMAND "STA/PASS:1/FLAGS:HOE". THIS WOULD BE A VERY TYPICAL WAY TO RUN THE DIAGNOSTIC. IF NO ERRORS ARE ENCOUNTERED, THE SINGLE REQUESTED PASS WILL BE EXECUTED AND THE PROMPT WILL BE RE-ISSUED.

IF AN ERROR IS ENCOUNTERED, THE ERROR WILL BE REPORTED AND THE PROMPT WILL BE REISSUED (BECAUSE THE HOE FLAG IS SET). AT THIS POINT THERE ARE FOUR DIFFERENT WAYS YOU CAN GET THE PROGRAM GOING AGAIN:

1. ISSUE ANOTHER "START" COMMAND (THUS GOING THRU ALL OF STEPS 1, 2, 3, 4, AND 5 AGAIN)
2. ISSUE A "RESTART" COMMAND (SAME AS START COMMAND EXCEPT THAT THE HARDWARE QUESTIONS ARE NOT ASKED)
3. ISSUE A "CONTINUE" COMMAND (EXECUTION WILL RESUME AT THE BEGINNING OF THE PARTICULAR HARDWARE TEST (MOST DIAGNOSTICS CONSIST OF A NUMBER OF THESE) THAT IT WAS IN WHEN THE ERROR HALT OCCURRED. NO QUESTIONS ASKED.
4. ISSUE A "PROCEED" COMMAND: EXECUTION WILL RESUME AT THE INSTRUCTION FOLLOWING THE ERROR REPORT (THIS IS A SPECIAL COMMAND AND CAN BE ISSUED ONLY AT A HALT

THE MOST TYPICAL THING TO DO HERE IS TO ISSUE THE PROCEED, BUT WITH DIFFERENT FLAG SETTINGS. PROBABLY YOU WOULD WANT TO SAY:

```
PRO/FLAGS:IER:LOE:HOE=0
```

THIS WILL DO THE FOLLOWING:

1. TURN ON THE IER (INHIBIT ERROR PRINTOUT) FLAG
2. TURN ON THE LOE FLAG
3. TURN OFF THE HOE FLAG
4. RESUME EXECUTION AT INSTRUCTION AFTER ERROR REPORT

THE DIAGNOSTIC WILL NOW LOOP ON THE BLOCK OF CODE THAT DETECTED AND REPORTED THE ERROR, BUT NO ERROR PRINTOUT WILL OCCUR. THUS YOU CAN STUDY THE ERROR OR SCOPE IT OR WHATEVER.

WHEN YOU'VE SEEN ENOUGH, YOU MAY HIT CONTROL/C. THIS WILL TAKE YOU OUT OF THE LOOP AND PUT YOU BACK INTO COMMAND MODE. YOU NOW HAVE THREE CHOICES:

1. START
2. RESTART
3. CONTINUE

LET'S SAY YOU'VE REPAIRED THE DEFECT FOUND ABOVE AND WANT TO FINISH RUNNING THE DIAGNOSTIC. YOU WOULD TYPE

```
CON/FLAGS:HOE:IER=0:LOE=0
```

THIS WILL RESTORE THE FLAGS TO THEIR ORIGINAL VALUES AND RESUME EXECUTION AT THE BEGINNING OF THE HARDWARE TEST YOU WERE IN. IF THE ERROR DOES NOT RECUR, THE EXECUTION WILL FLOW RIGHT ON THRU TO THE NEXT ERROR OR TO END OF PASS.

IF AT END OF PASS YOU WANT TO RUN THE DIAGNOSTIC AGAIN, YOU HAVE TWO CHOICES:

1. START
2. RESTART

YOU WOULD CHOOSE ONE, DEPENDING ON WHETHER YOU WANTED TO ANSWER THE HARDWARE QUESTIONS AGAIN.

THE FULL PRINT-OUT FROM THE ABOVE DIALOGUE MIGHT LOOK LIKE THIS
(O=OPERATOR, D=DIAGNOSTIC):

```
                                        BY
                                        WHOM
                                        ENTERED:
                                        -------

.R CZRLLC                               O
DRS LOADED                              D
DIAG. RUN-TIME SERVICES REV. C APR-79   D
CZRLL-B-0                               D
CZRLL VERIFIES INTERCHANGEABILITY OF    D
CARTRIDGES BETWEEN DRIVES
UNIT IS RL01, RL02                      D
DR>STA/PASS:1/FLAGS:HOE                 D,O

CHANGE HW (L) ? Y                       D,O

# UNITS (D) ? 2                        D,O

UNIT 0                                  D
BUS ADDRESS (O) 174400 ?                D,O
VECTOR (O) 160 ?                        D,O
DRIVE (O) 0 ?                           D,O
DRIVE TYPE = RL01 (L) Y ?               D,O

UNIT 1                                  D
BUS ADDRESS (O) 174400 ?                D,O
VECTOR (O) 160 ?                        D,O
DRIVE (O) 0 ? 1                         D,O
DRIVE TYPE = RL01 (L) ? N               D,O (N=RL02)

CZRLL HRD ERR 00004 TST 003 SUB 002 PC:004130
ERR HLT

DR>PRO/FLAGS:IER:LOE:HOE=0              D,O


**************************************************
AT THIS POINT THE DIAGNOSTIC IS LOOPING ON THE
ERROR WITHOUT PRINTING ANYTHING.  YOU CAN SCOPE
THE ERROR UNTIL YOU HAVE LOCATED IT, THEN ^C OUT.
**************************************************

^C                                      O

DR>CON/FLAGS:HOE:IER:LOE=0              D,O
```

```
CZRLL EOP 1                          D
^C

DR>RESTART/PASS:1                    D,O

-----------------
-----------------
-----------------
-----------------
```

2.2 CHAIN MODE OPERATION

NOT THIS PROGRAM IS NOT CHAINABLE. CHAIN MODE OPERATION CONSISTS OF THE SEQUENTIAL EXECUTION OF PROGRAMS WITHOUT OPERATOR INTERVENTION. ONLY PROGRAMS THAT HAVE BEEN MODIFIED TO RUN IN CHAIN MODE CAN BE CHAINED. CHAINABLE PROGRAMS ARE IDENTIFIED IN THE DIRECTORY BY A BIC EXTENSION.

TO RUN CHAIN MODE, THE XXDP+ MONITOR USES AN ASCII FILE (KNOWN AS A CHAIN FILE) LISTING THE PROGRAMS TO BE RUN AND THE NUMBER OF PASSES EACH PROGRAM SHOULD RUN. THIS FILE MUST BE ON THE SYSTEM DEVICE.

A CHAIN FILE MAY BE GENERATED BY USE OF THE XTECO TEXT EDITOR. THIS FILE MUST HAVE A CCC EXTENSION. THE CHAIN FILE MAY CONTAIN ANY OF THE COMMANDS SUPPORTED BY THE XXDP+ MONITOR. THE COMMANDS IN THE ASCII FILE ARE EXECUTED IN THE ORDER IN WHICH THEY ARE ENCOUNTERED.

TO EXECUTE A CHAIN FILE THE USER TYPES:

```
C FILNAM <CR> OR
C FILNAM/QV  <CR>
```

IN THE FIRST CASE THE PASS COUNT SPECIFIED IN THE CHAIN FILE IS USED BY THE XXDP+ MONITOR TO DETERMINE THE NUMBER OF PASSES TO EXECUTE EACH PROGRAM. IN THE SECOND CASE THE PASS COUNT IS NOT USED AND EACH PROGRAM IS EXECUTED ONLY ONCE. THE /QV SWITCH PROVIDES A SINGLE EXECUTION MODE OF OPERATION OF QUICK VERIFY.

WHEN PROGRAMS ARE RUN IN CHAIN MODE, THE SOFTWARE SWITCH REGISTER SHOULD BE SET TO 000000. THE XXDP+ MONITOR PRINTS EACH COMMAND TAKEN FROM THE CHAIN FILE AND THEN EXECUTES THE COMMAND. WHEN THE LAST COMMAND OTHER THAN ANOTHER C COMMAND HAS BEEN EXECUTED THE XXDP+ MONITOR TERMINATES CHAIN MODE AND TYPES A PROMPT (.). IT IS READY TO ACCEPT ANOTHER COMMAND FROM THE CONSOLE. IF THE LAST COMMAND IS ANOTHER C COMMAND, THE CHAIN MODE WILL CONTINUE AND THE CHAIN FILE SPECIFIED BY THIS NEW C COMMAND WILL BE USED.

IF THE USER WISHES TO TERMINATE CHAIN MODE BEFORE ITS NORMAL TERMINATION HE MAY DO SO BY TYPING A CONTROL/C. HOWEVER, THE MONITOR WILL NOT ABORT THE CHAIN MODE UNTIL IT RECEIVES PROGRAM CONTROL FROM THE PROGRAM CURRENTLY RUNNING.

2.3 DETAILS OF COMMANDS AND SYNTAX

2.3.1 TABLE OF COMMAND VALIDITY

THERE ARE FOUR WAYS OF ENTERING DIAGNOSTIC COMMAND MODE, AND DIFFERENT SUBSETS OF THE DIAG COMMAND SET ARE AVAILABLE WITH EACH:

HOW ENTERED		LEGAL COMMANDS
1.	OPERATOR ENTERED 'RUN DIAG'	START PRINT DISPLAY FLAGS ZFLAGS EXIT
2.	DIAGNOSTIC HAS FINISHED ALL ITS REQUESTED PASSES	START RESTART PRINT DISPLAY FLAGS ZFLAGS EXIT
3.	OPERATOR INTERRUPTED THE DIAGNOSTIC WITH CTRL/C	START RESTART CONTINUE PRINT DISPLAY FLAGS ZFLAGS EXIT
4.	AN ERROR WAS ENCOUNTERED WITH THE HOE FLAG SET SET	START RESTART CONTINUE PROCEED PRINT DISPLAY FLAGS ZFLAGS EXIT

2.3.2 COMMAND SYNTAX

****
STA(RT)/TESTS:TEST-LIST/PASS:PASS-CNT/FLAGS:FLAG-LIST/EOP:EOP-INCR
****

THE DIAGNOSTIC IN CORE IS EXECUTED IN ACCORDANCE WITH THE SWITCHES SPECIFIED. THE MESSAGE "# UNITS?" IS PRINTED. THE START COMMAND MAY BE ISSUED WHEN DIAGNOSTIC COMMAND MODE HAS BEEN ENTERED VIA ONE OF THE FOLLOWING: A) OPERATOR TYPED "RUN DIAGNOSTIC" B) DIAGNOSTIC FINISHED EXECUTING C) ERROR WAS ENCOUNTERED WITH HOE FLAG SET D) OPERATOR ENTERED CONTROL/C. AFTER THE OPERATOR RESPONDS TO "# UNITS?", THE HARDWARE DIALOGUE IS INITIATED. WHEN IT IS COMPLETED, THE QUESTIONS "CHANGE SW?" IS ISSUED, AND THE ANSWERS, IF GIVEN, BECOME THE NEW DEFAULTS. THEREFORE IT IS NECESSARY TO RELOAD THE PROGRAM IN ORDER TO RETURN TO THE LOAD DEFAULTS.

THE SWITCH ARGUMENTS ARE AS FOLLOWS:

"TEST-LIST" IS A SEQUENCE OF DECIMAL NUMBERS (1:2 ETC.) OR RANGES OF DECIMAL NUMBERS (1-5:8-10 ETC.) THAT SPECIFY THE TESTS TO BE EXECUTED. THE NUMBERS ARE SEPARATED BY COLONS. THE NUMBERS RANGE FROM 1 TO THE LARGEST TEST NUMBER IN THE DIAGNOSTIC. THEY MAY BE SPECIFIED IN ANY ORDER. TESTS WILL BE EXECUTED IN NUMERICAL ORDER REGARDLESS OF THE ORDER OF SPECIFICATION. THE DEFAULT IS TO EXECUTE ALL TESTS.

"PASS-CNT" IS A DECIMAL NUMBER INDICATING THE DESIRED NUMBER OF PASSES. A PASS IS DEFINED AS THE EXECUTION OF THE FULL DIAGNOSTIC (ALL SELECTED TESTS) AGAINST ALL UNITS SUBMITTED. THE DEFAULT IS NON-ENDING TEST EXECUTION. "FLAG-LIST" IS A SEQUENCE OF ELEMENTS OF THE FORM <FLAG>, <FLAG=1>, OR <FLAG=0>, SEPARATED BY COLONS, WHERE <FLAG> HAS ONE OF THE FOLLOWING VALUES:

HOE HALT ON ERROR, CAUSING COMMAND MODE TO BE ENTERED WHEN AN ERROR IS ENCOUNTERED

LOE LOOP ON ERROR, CAUSING THE DIAGNOSTIC TO LOOP CONTINUOUSLY WITHIN THE SMALLEST DEFINED BLOCK OF CODING (SEGMENT, SUBTEST, OR TEST) CONTAINING THE ERROR

IER INHIBIT ERROR REPORTING

IBE INHIBIT BASIC ERROR REPORTS

IXE INHIBIT EXTENDED ERROR REPORTS

PRI DIRECT ALL MESSAGES TO A LINE PRINTER

PNT PRINT NUMBER OF TEST BEING EXECUTED

BOE BELL ON ERROR

UAM RUN IN UNATTENDED MODE, BYPASSING MANUAL INTERVENTION TESTS

ISR INHIBIT STATISTICAL REPORTS

IDU INHIBIT DROPPING OF UNITS BY DIAGNOSTIC

ADR EXECUTE AUTODROP CODE

LOT LOOP ON TEST

EVL EVALUATE

THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO 0 ARE CLEARED. A FLAG NOT SPECIFIED IS CLEARED. IF THE FLAGS SWITCH IS NOT GIVEN ALL FLAGS ARE CLEARED.

"EOP-INCR" IS A DECIMAL NUMBER INDICATING HOW OFTEN (IN TERMS OF PASSES) IT IS DESIRED THAT THE END OF PASS MESSAGE BE PRINTED. THE DEFAULT IS AT THE END OF EVERY PASS.

****

RES(TART)/TEST:TEST-LIST/PASS:PASS-CNT/FLAGS:FLAG-LIST/EOP:EOP-INCR/UNITS:UNIT-LIST

****

THE DIAGNOSTIC IN CORE IS EXECUTED IN ACCORDANCE WITH THE SWITCHES SPECIFIED. HOWEVER, NEW "P-TABLES" ARE NOT BUILT. INSTEAD, THE ONES IN CORE ARE USED.

THE QUESTION "CHANGE SW?" IS ASKED AND THE ANSWERS GIVEN BECOME THE NEW DEFAULTS. THE COMMAND MAY BE ISSUED WHEN COMMAND MODE HAS BEEN ENTERED VIA A) DIAGNOSTIC IS FINISHED B) HALT ON ERROR C) CONTROL/C.

THE SWITCH ARGUMENTS ARE AS IN THE START COMMAND EXCEPT:

1. "UNIT-LIST" IS A SEQUENCE OF LOGICAL UNIT NUMBERS RANGING FROM 1 THRU N (N = NUMBER OF UNITS BEING TESTED) SPECIFYING WHICH UNITS ARE TO BE TESTED. THE LOGICAL UNIT NUMBER DESIGNATES THE POSITION OF THE P-TABLE IN CORE, ACCORDING TO THE ORDER IN WHICH THEY WERE BUILT. THE UNITS SPECIFIED MUST NOT HAVE BEEN DROPPED BY THE OPERATOR DROP COMMAND. THE UNIT-LIST DEFAULTS TO "ALL THAT HAVE NOT BEEN DROPPED BY OPERATOR COMMAND". THE EFFECT OF THE UNIT-LIST LASTS UNTIL THE NEXT START (WHERE IT IS AUTOMATICALLY RESET TO "ALL") OR THE NEXT RESTART.

2. ALL UNSPECIFIED FLAG SETTINGS ARE UNCHANGED.

****
CON(TINUE)/PASS:<PASS-CNT/FLAGS:<FLAG-LIST>
****

COMMAND MODE MUST HAVE BEEN ENTERED DUE TO A HALT ON ERROR OR A CONTROL/C. THE EFFECT OF THE COMMAND IS TO GO TO THE BEGINNING OF THE TEST THAT WAS BEING EXECUTED WHEN THE HALT OR CONTROL/C TOOK PLACE. SOFTWARE DIALOGUE MAY OPTIONALLY BE RE-EXECUTED. HARDWARE PARAMETERS MAY NOT BE CHANGED.

THE SWITCH ARGUMENTS ARE AS IN THE START COMMAND EXCEPT:

1. DEFAULT FOR PASS-CNT IS THE UNSATISFIED PASS-CNT FROM THE PREVIOUS START OR RESTART

2. UNSPECIFIED FLAG SETTINGS ARE UNCHANGED

****
PRO(CEED)/FLAGS:<FLAG-LIST>
****

COMMAND MODE MUST HAVE BEEN ENTERED VIA A HALT ON ERROR. THE EFFECT OF THE COMMAND IS TO BEGIN EXECUTION AT THE LOCATION FOLLOWING THE ERROR CALL. NEITHER HARDWARE NOR SOFTWARE PARAMETERS MAY BE ALTERED.

THE SWITCH ARGUMENTS ARE THE SAME AS THE START COMMAND EXCEPT:

1. UNSPECIFIED FLAG SETTINGS ARE UNCHANGED

****
EXIT
****

RETURN TO XXDP+ PROMPT MODE.

****
DRO(P)/UNITS:UNIT-LIST
****

THE UNITS SPECIFIED ARE DROPPED FROM TESTING UNTIL THEY ARE ADDED BACK OR UNTIL A START COMMAND IS GIVEN. A DROP CANNOT BE FOLLOWED BY A PROCEED.

THERE IS ALSO A "DROP" MACRO INTERNAL TO THE DIAGNOSTIC, WHICH GIVES THE FACILITY OF AUTO-DROPPING. THE DURATION OF A PROGRAM DROP, HOWEVER, IS ONLY UNTIL THE NEXT START OR RESTART.

****
ADD/UNITS:UNIT-LIST
****

THE UNITS SPECIFIED ARE ADDED BACK (THEY MUST HAVE BEEN PREVIOUSLY DROPPED BY THE DROP COMMAND) TO THE TEST SEQUENCE. AN ADD CANNOT BE FOLLOWED BY A PROCEED.

****
PRI(NT)
****

ALL STATISTICS TABLES ACCUMULATED BY THE DIAGNOSTIC ARE PRINTED. THE ISR (INHIBIT STATISTICAL REPORTING) FLAG IS CLEARED.

****
DIS(PLAY)/UNITS:<UNIT-LIST>
****

THE HARDWARE P-TABLES FOR ALL UNITS UNDER TEST ARE PRINTED OUT IN THE FORMAT IN WHICH THEY WERE ENTERED. ANY UNITS THAT WERE DROPPED BY THE OPERATOR "DROP" COMMAND ARE SO DESIGNATED.

****
FLA(GS)
****

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

****
ZFL(AGS)
****

ALL FLAGS ARE CLEARED.

2.4 EXTENDED P-TABLE DIALOGUE

THE FULL CAPABILITY OF THE HARDWARE DIALOGUE IS REVEALED BY THE FOLLOWING DISCUSSION OF WHAT HAPPENS INTERNALLY.

AS SCON AS THE QUESTION "# UNITS?" IS ANSWERED (WITH THE NUMBER N), SPACE IN CORE IS ALLOCATED FOR "N" P-TABLES. ALL OF THE P-TABLES ARE OF THE SAME FORMAT, AND THERE IS A ONE-TO-ONE CORRESPONDENCE BETWEEN THE HARDWARE PARAMETER QUESTIONS AND THE SLOTS IN THE P-TABLE FORMAT.

IN GIVING A STRING OF VALUES, COMMAS WITHOUT INTERVENING VALUES MAY BE USED TO INDICATE A REPETITION OF THE LAST NAMED VALUE.

A STRING OF VALUES MAY BE GIVEN AS A RANGE (6-10 FOR EXAMPLE). IF THE VALUES REPRESENT PURE NUMERICAL DATA, THIS SAMPLE RANGE TRANSLATES TO THE STRING 6,7,8,9,10 (AN INCREMENT OF 1). IF THE VALUES ARE ADDRESSES, THE SAMPLE RANGE TRANSLATES TO THE STRING 6,8,10 (AN INCREMENT OF 2).

NOW LET US SEE HOW WE COULD USE THESE CAPABILITIES TO CONSTRUCT A SET OF P-TABLES. ASSUME THAT WE HAVE 8 RL UNITS, AND THAT THERE ARE FIVE (5) HARDWARE PARAMETERS FOR EACH (5 SLOTS IN THE P-TABLE, 5 HARDWARE QUESTIONS IN THE DIALOGUE).

FOLLOWING IS THE DIALOGUE FOR THIS 8 RLOX DRIVE SYSTEM. THIS SYSTEM HAS TWO (2) RL11 TYPE CONTROLLERS ALL TO BE SET AT 'BR LEVEL' 5. THE FIRST 4 DRIVES ARE RL01'S AND THE LAST 4 DRIVES ARE RL02'S (ON THE SECOND CONTROLLER):

```
  UNITS (D) ? 8

UNIT 0
BUS ADDRESS (O) 174400 ?
VECTOR (O) 160 ?
DRIVE (O) 0 ? 0-3
DRIVE TYPE = RL01 (L) Y ?

UNIT 4
BUS ADDRESS (O) 174400 ? 175400
VECTOR (O) 160 ? 164
DRIVE (O) 0 ? 0-3
DRIVE TYPE = RL01 (L) Y ? N
```

THE FIRST TIME THRU THE P-TABLE QUESTIONS THE DEFAULT VALUES ARE USED FOR THE CSR ADDRESS OF THE CONTROLLER (QUESTION #1), THE CONTROLLER VECTOR ASSIGNMENT (QUESTION #2), AND THE DRIVE TYPE (QUESTION #4). THE ACTUAL UNIT NUMBERS OF THE RL01'S FOR QUESTION #3 WAS ASSIGNED 0 THRU 3 FOR THE FIRST 4 P-TABLE SLOTS.

THE SECOND TIME THRU THE P-TABLE QUESTIONS THE FIRST QUESTION WAS ANSWERED TO REFLECT THE CHANGE IN CSR ADDRESS FOR THE RL02 CONTROLLER (175400). THE SECOND CONTROLLER'S VECTOR WAS ALSO CHANGED TO 164 IN QUESTION #2. THE RL02 TEST UNIT NUMBERS WERE ASSIGNED VALUES 0 TO 3 IN QUESTION #3 AND THE DRIVE TYPE WAS SET FOR RL02'S FOR THE REMAINING 4 UNITS IN QUESTION #4.

2.5 HARDWARE PARAMETERS

THE FOLLOWING QUESTIONS WILL BE ASKED ON A START COMMAND. THE VALUE LOCATED TO THE LEFT OF THE QUESTION MARK IS THE DEFAULT VALUE THAT WILL BE TAKEN ON A CARRIAGE RETURN RESPONCE.

BUS ADDRESS (O) 174400?

ANSWER WITH THE BUS ADDRESS OF THE CONTROLLER.

VECTOR (O) 160?

ANSWER WITH THE INTERRUPT VECTOR OF THE CONTROLLER.

DRIVE (O) 0?

ANSWER WITH THE DRIVE(S) CONNECTED TO THE CONTROLLER

DRIVE TYPE = RL01 (L) ?

ANSWER NO (N) IF DRIVE IS AN RL02

2.6 SOFTWARE PARAMETERS

THE FOLLOWING QUESTIONS ARE ASKED IF REQUESTED ON A START, RESTART, OR CONTINUE. THEY ALLOW FLEXABILITY IN THE WAY THE PROGRAM BEHAVES. THE SOFTWARE PARAMETERS GIVE THE PROGRAM FLEXIBILITY IN THE WAY IT RUNS. THE PARAMETERS CAN BE MODIFIED ON A START, RESTART, OR CONTINUE BY ANSWERING (Y)ES TO THE FOLLOWING QUESTION:

"CHANGE S.W. ?"

A YES ANSWER WILL ASK THE FOLLOWING SOFTWARE PARAMETER QUESTIONS, WITH THE PRESENT DEFAULT VALUE PRINTED TO THE LEFT OF THE QUESTION MARK. (THE LAST ANSWER GIVEN IS THE DEFAULT) THE DEFAULT IS TAKEN ON A <CR>. CONTROL Z (^Z) WILL DEFAULT ALL REMAINING QUESTIONS AND START THE TEST.

THERE ARE NO SOFTWARE PARAMETERS.

3.0 ERROR INFORMATION

ERROR INFORMATION IS COMPLETE IN GIVING ALL INFORMATION NECESSARY. ALL REGISTERS ARE GIVEN AS WELL AT TRACK, SECTOR AND DRIVES INVOLVED IN ERROR.

3.1 ERROR REPORTING

ALL ERROR INFORMATION IS PRINTED ON THE CONSOLE DEVICE. ERROR REPORTS ARE AIMED AT BEING SELF EXPLANATORY. THE GENERAL FORMAT IS:

```
DZRLL  XXX    ERR  YYYYY   TST  ZZZ   SUB  PPP   PC:  RRRRRR
```

WHERE:

?	IS PROGRAM LETTER
XXX	IS SFT - SOFT ERROR
	HRD - HARD ERROR
	DV FAT - DEVICE FATAL ERROR
	SYS FAT - SYSTEM FATAL ERROR
YYYYY	IS THE ERROR NUMBER
ZZZ	IS THE TEST NUMBER
PPP	IS THE SUBTEST NUMBER
RRRRRR	IS THE PROGRAM LISTING LOCATION

ERRORS GIVE THE REGISTER CONTENTS BEFORE AND AFTER THE ERROR ALONG WITH A ONE LINE DESCRIPTION AND RELEVENT DATA.

EXAMPLE:

```
ONE LINE DESCRIPTION
(OPTIONAL SECOND LINE)
(OPTIONAL THIRD LINE)
BEFORE CS:XXXXXX BA:XXXXXX DA:XXXXXX MP:XXXXXX
AFTER  CS:XXXXXX BA:XXXXXX DA:XXXXXX MP:XXXXXX
OTHER PERTINENT INFORMATION IS GIVEN AT THIS TIME.
```

REGISTER DESCRIPTIONS CAN BE FOUND IN SECTION 5.0.

ERROR DESCRIPTIONS:

"ERROR READING SECTOR"

ERROR WAS ENCOUNTERED WHILE TRYING TO READ VERIFY THE SECTOR AFTER IT WAS WRITTEN BY THE SAME DRIVE.

"MINIMUM OF TWO DRIVES REQUIRED"

THE PROGRAM REQUIRES AT LEAST TWO DRIVES TO PROVE COMPATABILITY.

''MAXIMUM OF FOUR DRIVES ALLOWED''

THE PROGRAM ONLY ALLOWS A MAXIMUM OF FOUR DRIVES.

''CAN'T FIND FIVE ADJACENT TRACKS''

THE PROGRAM REQUIRES TEN SETS OF FIVE ADJACENT TRACKS AT PREDETERMINED SPOTS ACROSS THE PACK. IT WAS UNABLE TO FIND FIVE COMPLETELY GOOD ADJACENT TRACKS IN THE LIMITS GIVEN.

''ERROR WRITING SECTOR''

AN ERROR WAS ENCOUNTERED WHILE TRYING TO WRITE THE GIVEN SECTOR.

''OVERWRITE ERROR''

AN ERROR WAS ENCOUNTERED WHILE TRYING TO READ DATA AFTER AN OVERWRITE BY ONE DRIVE. BOTH DRIVES INVOLVED ARE GIVEN.

''READ RECOVERY ERROR''

AN ERROR WAS ENCOUNTERED WHILE TRYING TO RECOVER ANOTHER DRIVES DATA.

''ADJACENT TRACK TEST''

AN ERROR WAS ENCOUNTERED WHILE IN THE ADJACENT TEST PART, A FURTHER DESCRIPTION IS GIVEN.

3.2 ERROR HALTS

ERROR HALTS ARE SUPPORTED PER DESCRIBED IN THE PREVIOUS SECTION WITH /FLAG:HOE. THERE ARE NO OTHER HALTS.

4.0 PERFORMANCE AND PROGRESS REPORTS

4.1 PERFORMANCE REPORTS

THIS PROGRAM WILL NOT GIVE ANY PERFORMANCE REPORTS.

4.2 PROGRESS REPORTS

THIS PROGRAM WILL NOT GIVE ANY PROGRESS REPORTS.

5.0 DEVICE INFORMATION TABLES

THE RL11/RLV11 CONTROLLER HAS THE FOLLOWING FOUR(4) REGISTERS FOR CONTROL OF THE SUBSYSTEM.

RLCS - CONTROL AND STATUS REGISTER (XXXXX0)

BIT 15 - COMPOSITE ERROR
BIT 14 - DRIVE ERROR
BIT 13 - NON EXISTANT MEMORY ERROR
BIT 12 - HEADER NOT FOUND (WITH BIT 10 SET)
- DATA LATE (WITH BIT 10 CLEAR)
BIT 11 - HEADER CRC (WITH BIT 10 SET)
- DATA CRC (WITH BIT 10 CLEAR)
BIT 10 - OPERATION INCOMPLETE
BIT 9/8 - DRIVE SELECT (0-3)
BIT 7 - CONTROLLER READY
BIT 6 - INTERRUPT ENABLE
BIT 5 - EXTENDED BUS ADDRESS (BIT 17)
BIT 4 - EXTENDED BUS ADDRESS (BIT 16)
BIT 3-1 - FUNCTION CODE
0 - NOP (PDP-11) MAINT (LSI-11)
1 - WRITE CHECK
2 - GET DRIVE STATUS
3 - SEEK
4 - READ HEADER
5 - WRITE DATA
6 - READ DATA
7 - READ WITHOUT HEADER COMPARE

BIT 0 - DRIVE READY

RLBA - BUS ADDRESS REGISTER (XXXXX2)

BITS 15-1 BUS ADDRESS OF DATA TRANSFER
BIT 0 SHOULD BE 0

RLDA - DISK ADDRESS REGISTER (XXXXX4)

FOR READ/WRITE FUNCTIONS

BIT 15-7 - CYLINDER ADDRESS FOR TRANSFER
BIT 6 - SURFACE FOR TRANSFER
BIT 5-0 - SECTOR FOR TRANSFER (1-40.)

FOR SEEK FUNCTION

BIT 15-7 - DIFFERENCE TO NEW CYLINDER
BIT 6-5 - MUST BE ZERO (0)
BIT 4 - SURFACE (0=UPPER, 1=LOWER)
BIT 3 - MUST BE ZERO (0)
BIT 2 - SEEK DIRECTION(1=IN / 0=OUT)
BIT 1 - MUST BE ZERO (0)
BIT 0 - MUST BE ONE (1)

FOR GET STATUS FUNCTION

BIT 15-4 - IGNORED SHOULD BE ZERO (0)
BIT 3 - DRIVE RESET
BIT 2 - MUST BE ZERO (0)
BIT 1 - MUST BE ONE (1)
BIT 0 - MUST BE ONE (1)

RLMP - MULTIPURPOSE REGISTER

FOR READ/WRITE FUNCTION

BIT 15 - 0 - WORD COUNT (TWO'S COMPLIMENT)

FOR READ HEADER FUNCTION

BIT 15-0 - DISK HEADER OF SECTOR (FIRST READ)
- ZERO WORD (SECOND READ)
- HEADER CRC (THIRD READ)

FOR GET STATUS FUNCTION

HAS DRIVE STATUS

BIT 15 - WRITE DATA ERROR
BIT 14 - CURRENT HEAD ERROR (CHE)
BIT 13 - WRITE LOCK STATUS (WL)
BIT 12 - SEEK TIME OUT (SKTO)
BIT 11 - SPIN ERROR (SPE)
BIT 10 - WRITE GATE ERROR (WGE)

BIT 9 - VOLUME CHECK (VC)
BIT 8 - DRIVE SELECT ERROR (DSE)
BIT 7 - DRIVE TYPE IS RL02 IF SET
BIT 6 - SURFACE (0=UPPPER, 1=LOWER)
BIT 5 - COVER OPEN
BIT 4 - HEADS HOME
BIT 3 - BRUSHES HOME
BIT 2-0 -STATE BITS
0 - LOAD STATE
1 - SPIN UP
2 - BRUSH CYCLE
3 - LOAD HEADS
4 - SEEK - TRACK COUNTING
5 - SEEK - LINEAR MODE
6 - UNLOAD HEADS
7 - SPIN DOWN

6.0 TEST SUMMARIES

THE FOLLOWING IS A BRIEF DESCRIPTION OF THE WAY THE PROGRAM EXECUTES. THE PROGRAM WILL CHECK COMPATIBILITY BETWEEN 2 - 4 DRIVES USING THE SAME RL01K CARTRIDGE OR SAME RL02K CARTRIDGE. THE PROGRAM WILL ASK THE OPERATOR TO SEQUENCE THE PACK BETWEEN THE DRIVES GIVEN IN THE FOLLOWING MANNER.

PLACE PACK IN DRIVE N ON CONTROLLER X AND LOAD

UNLOAD DRIVE N ON CONTROLLER X

PLACE PACK IN DRIVE N+1 ON CONTROLLER X AND LOAD

UNLOAD DRIVE N+1 ON CONTROLLER X

ETC.....

THE PROGRAM WILL SEQUENCE IN THE ORDER THAT WAS GIVEN IN THE HARDWARE QUESTIONS. I.E.

DRIVE ? 0,1,2,3

PROGRAM WILL SEQUENCE 0,1,2,3,2,1,0

DRIVE ? 1,0,3,2

PROGRAM WILL SEQUENCE 1,0,3,2,3,0,1

WHEN THE FIRST DRIVE IS LOADED THE PROGRAM WILL ATTEMPT TO FIND TEN SETS OF FIVE ADJACENT TRACKS AT PREDETERMINED SPOTS THAT CONTAIN NO BAD SECTORS USING THE BAD SECTOR FILE. THE 10 SPOTS ARE: ON BOTH SURFACES, INNER, OUTER, MIDDLE, ONE QUARTER AND THREE QUARTERS. AFTER THIS IS DONE THE OVERWRITE TEST IS PREPARED(FIRST DRIVE CAN'T OVERWRITE) AS WELL AS THE ADJACENT TEST. AS THE PACK IS CYCLED BETWEEN DRIVES THE FOLLOWING CHECKS ARE MADE:

EACH DRIVE CAN OVERWRITE EACH OTHER DRIVE

EACH DRIVE CAN RECOVER EACH OTHERS DATA

EACH DRIVE CAN WRITE ADJACENT TO EVERY OTHER DRIVE WITHOUT DISTURBING THE OTHER'S DATA.

READS AND WRITES TAKE PLACE AFTER SEEKS FROM BOTH DIRECTIONS.

ADJACENT WRITES TAKE PLACE TO BOTH SIDES OF EACH WRITE

TESTS ARE PERFORMED AT ALL TEN SPOTS ACROSS THE PACK.

a

CZRLLCO RL01/02 DRIVE COMPAT MACRO V04.00 16-FEB-82 13:32:06
TABLE OF CONTENTS

```
                                .TITLE  CZRLLCO RL01/02 DRIVE COMPAT
                                .ENABLE AMA
000000                          .ENABLE ABS
                                .MCALL  SVC
002000                          .=2000


                                .SBTTL  MACRO DEFINITIONS

                                .MACRO  WAITUS  ARG             ;MACRO  MICRO-SECOND WAIT
                                        MOV     ARG,XDELAY      ;SAVE ARGUMENT
                                        JSR     PC,TIME         ;CALL TIMING ROUTING
                                .ENDM

                                .MACRO  WAITMS  ARG             ;MACRO MILLI-SECOND WAIT
                                        MOV     ARG,YDELAY      ;SAVE ARGUMENT
                                        JSR     PC,XTIME        ;CALL TIMING ROUTINE
                                .ENDM

                                .NLIST  CND,MD,ME

002000                          SVC
        000000                  SVCINS=0
        000000                  SVCTAG=0

002000                                  POINTER NONE

002000                          BGNMOD  MDHEDR
002000                                  HEADER  CZRLL,C,0,0,1
002000      103                         .ASCII /C/
002001      132                         .ASCII /Z/
002002      122                         .ASCII /R/
002003      114                         .ASCII /L/
002004      114                         .ASCII /L/
002005      000                         .BYTE   0
002006      000                         .BYTE   0
002007      000                         .BYTE   0
002010      103                         .ASCII  /C/
002011      060                         .ASCII  /0/
002012   000000                         .WORD   0
002014   000000                         .WORD   0
002016   033652                         .WORD   L$HARD
002020   000000                         .WORD   0
002022   022450                         .WORD   L$HW
002024   000000                         .WORD   0
002026   034014                         .WORD   L$LAST
002030   000000                         .WORD   0
002032   000000                         .WORD   0
002034   000001                         .WORD   1
002036   000000                         .WORD   0
002040   022464                         .WORD   L$DISPATCH
002042   000000                         .WORD   0
002044   000000                         .WORD   0
002046   000000                         .WORD   0
002050      003                         .BYTE   C$REVISION
002051      003                         .BYTE   C$EDIT
002052   000000                         .WORD   0
```

MACRO DEFINITIONS

```
002054  000000                          .WORD   0
002056  000000                          .WORD   0
002060  002222                          .WORD   L$DVTYP
002062  000000                          .WORD   0
002064  000000                          .WORD   0
002066  000000                          .WORD   0
002070  000000                          .WORD   0
002072  000000                          .WORD   0
002074  000000                          .WORD   0
002076  002122                          .WORD   L$DESC
002100  104035                          EMT     E$LOAD
002102  000000                          .WORD   0
002104  022466                          .WORD   L$INIT
002106  024316                          .WORD   L$CLEAN
002110  024312                          .WORD   L$AUTO
002112  022440                          .WORD   L$PROT
002114  000000                          .WORD   0
002116  000000                          .WORD   0
002120  000000                          .WORD   0
 30 002122                              ENDMOD
 31
 32 002122                      DESCRIPT        <CZRLL VERIFIES INTERCHANGEABILITY OF CARTRIDGES BETWEEN DRIVES>
    002122  103  132  122              .ASCIZ  /CZRLL VERIFIES INTERCHANGEABILITY OF CARTRIDGES BETWEEN DRIVES/
    002125  114  114  040
    002130  126  105  122
    002133  111  106  111
    002136  105  123  040
    002141  111  116  124
    002144  105  122  103
    002147  110  101  116
    002152  107  105  101
    002155  102  111  114
    002160  111  124  131
    002163  040  117  106
    002166  040  103  101
    002171  122  124  122
    002174  111  104  107
    002177  105  123  040
    002202  102  105  124
    002205  127  105  105
    002210  116  040  104
    002213  122  111  126
    002216  105  123  000
                                    .EVEN
 33
 34 002222                      DEVTYP          <RL01,RL02>
    002222  122  114  060           .ASCIZ  /RL01,RL02/
    002225  061  054  122
    002230  114  060  062
    002233  000
                                    .EVEN
 35
 36                             .SBTTL GLOBAL EQUATES SECTION
 37
 38                             ;DEFINITIONS
 39
 40 002234                      BGNMOD  GLBEQAT
```

```
GLOBAL EQUATES SECTION

41
42 002234                              EQUALS
                                       ;
                                       ; BIT DIFINITIONS
                                       ;
                100000                 BIT15== 100000
                040000                 BIT14== 40000
                020000                 BIT13== 20000
                010000                 BIT12== 10000
                004000                 BIT11== 4000
                002000                 BIT10== 2000
                001000                 BIT09== 1000
                000400                 BIT08== 400
                000200                 BIT07== 200
                000100                 BIT06== 100
                000040                 BIT05== 40
                000020                 BIT04== 20
                000010                 BIT03== 10
                000004                 BIT02== 4
                000002                 BIT01== 2
                000001                 BIT00== 1
                                       ;
                001000                 BIT9==  BIT09
                000400                 BIT8==  BIT08
                000200                 BIT7==  BIT07
                000100                 BIT6==  BIT06
                000040                 BIT5==  BIT05
                000020                 BIT4==  BIT04
                000010                 BIT3==  BIT03
                000004                 BIT2==  BIT02
                000002                 BIT1==  BIT01
                000001                 BIT0==  BIT00
                                       ;
                                       ; EVENT FLAG DEFINITIONS
                                       ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                       ;
                000040                 EF.START==      32.                     ; START COMMAND WAS ISSUED
                000037                 EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
                000036                 EF.CONTINUE==   30.                     ; CONTINUE COMMAND WAS ISSUED
                000035                 EF.NEW==        29.                     ; A NEW PASS HAS BEEN STARTED
                000034                 EF.PWR==        28.                     ; A POWER-FAIL/POWER-UP OCCURRED
                                       ;
                                       ;
                                       ; PRIORITY LEVEL DEFINITIONS
                                       ;
                000340                 PRI07== 340
                000300                 PRI06== 300
                000240                 PRI05== 240
                000200                 PRI04== 200
                000140                 PRI03== 140
                000100                 PRI02== 100
                000040                 PRI01== 40
                000000                 PRI00== 0
                                       ;
                                       ;OPERATOR FLAG BITS
                                       ;
                000004                 EVL==       4
```

GLOBAL EQUATES SECTION

```
000010                  LOT==       10
000020                  ADR==       20
000040                  IDU==       40
000100                  ISR==      100
000200                  UAM==      200
000400                  BOE==      400
001000                  PNT==     1000
002000                  PRI==     2000
004000                  IXE==     4000
010000                  IBE==    10000
020000                  IER==    20000
040000                  LOF==    40000
100000                  HOE==   100000

000000                  CS=0                    ;CONTROL AND STATUS OFFSET
000002                  BA=2                    ;BUSS ADDRESS OFFSET
000004                  DA=4                    ;DISK ADDRESS OFFSET
000006                  MP=6                    ;MULTI PURPOSE OFFSET

                        ;CONSTANT OFFSETS FOR INDIVIDUAL DRIVE BUFFERS

000000                  CSR=0                   ;CONTROLLER ADDRESS
000002                  VEC=2                   ;VECTOR OF CONTROLLER
000004                  DSB=4                   ;DRIVE SELECT
000006                  PAT=6                   ;PATTERN UNIQUE TO DRIVE


000001                  DRDY=BIT0               ;DRIVE READY
000100                  INTEN=BIT6              ;INTERRUPT ENABLE
100000                  ERR=BIT15               ;COMPOSITE ERROR
040000                  DERR=BIT14              ;DRIVE EPROR
020000                  NXM=BIT13               ;NON-EXISTANT MEMORY ERROR
010000                  DLT=BIT12               ;DATA LATE
004000                  DCRC=BIT11              ;DATA CRC ERROR
004000                  HCRC=BIT11              ;HEADER CRC ERROR
010000                  HNF=BIT12               ;HEADER NOT FOUND ERROR
002000                  OPI=BIT10               ;OPERATION INCOMPLETE ERROR
000200                  CRDY=BIT7               ;CONTROLLER READY
000040                  BA17=BIT5               ;EXTENDED BUS ADDRESS BIT 17
000020                  BA16=BIT4               ;EXTENDED BUS ADDRESS BIT 16
000002                  CRSET=BIT1              ;CONTROLLER RESET FUNCTION CODE
000004                  GSTAT=BIT2              ;GET DRIVE STATUS FUNCTION CODE
000006                  SEEK=BIT1!BIT2          ;SEEK FUNCTION CODE
000010                  RDHDR=BIT3              ;READ HEADER FUNCTION CODE
000012                  WRITE=BIT3!BIT1         ;WRITE FUNCTION CODE
000014                  READ=BIT3!BIT2          ;READ FUNCTION CODE
000013                  DRST=BIT3!BIT1!BIT0     ;DRIVE RESET COMMAND CODE FOR DRIVE COMMAD WORD
000003                  GSBIT=BIT1!BIT0         ;GET STATUS COMMAND CODE FOR DRIVE COMMAND WORD
000001                  MK=BIT0                 ;MARKER BIT FOR DRIVE COMMAND WORD(SEEK,GET STATUS)
000004                  SIGN=BIT2               ;DIRECTION FOR SEEK(0=AWAY FROM SPINDLE)
000020                  SKHS=BIT4               ;HEAD SELECT FOR SEEK
000100                  HEAD=BIT6               ;HEAD SELECT FOR READ,WRITE,GET STATUS

                        ;OFFSET FOR HARDWARE P-TABLE

000000                  CSR=    0               ;BUS ADDRESS
000002                  VECT=   2               ;VECTOR ADDRESS
```

GLOBAL EQUATES SECTION

```
     87          000004                  PRIOR=  4               ;PRIORITY (BREAK LEVEL)
     88          000006                  TYPDR=  6               ;DRIVE TYPE
     89          000010                  DRBT=   10              ;DRIVE SELECT
     90
     91 002234                                   ENDMOD
     92
     93
     94
```

```
                                .SBTTL GLOBAL DATA SECTION

002234                          BGNMOD  GLBDAT

002234  000000                  HDRFND: .WORD   0               ;1=HEADER IN BAD SECTOR LIST

                                ;HERE IS THE LIST OF TRACKS TO USE FOR THIS TEST
                                ;TRACKS ARE ENTERED BY 'FNDTRK' ROUTINE & 'FIXTRK' ROUTINE

002236  000000                  OUT10:  .WORD   0       ;OUTER TRK HEAD 0
002240  000000                  OUT20:  .WORD   0
002242  000000                  OUT30:  .WORD   0
002244  000000                  OUT40:  .WORD   0
002246  000000                  OUT50:  .WORD   0
002250  000000                  OUT11:  .WORD   0       ;OUTER TRK HEAD 1
002252  000000                  OUT21:  .WORD   0
002254  000000                  OUT31:  .WORD   0
002256  000000                  OUT41:  .WORD   0
002260  000000                  OUT51:  .WORD   0
002262  000000                  QQU10:  .WORD   0       ;1ST QUARTER TRK HEAD 0
002264  000000                  QQU20:  .WORD   0
002266  000000                  QQU30:  .WORD   0
002270  000000                  QQU40:  .WORD   0
002272  000000                  QQU50:  .WORD   0
002274  000000                  QQU11:  .WORD   0       ;1ST QUARTER TRK HEAD 1
002276  000000                  QQU21:  .WORD   0
002300  000000                  QQU31:  .WORD   0
002302  000000                  QQU41:  .WORD   0
002304  000000                  QQU51:  .WORD   0
002306  000000                  MID10:  .WORD   0       ;MIDDLE TRK HEAD 0
002310  000000                  MID20:  .WORD   0
002312  000000                  MID30:  .WORD   0
002314  000000                  MID40:  .WORD   0
002316  000000                  MID50:  .WORD   0
002320  000000                  MID11:  .WORD   0       ;MIDDLE TRK HEAD 1
002322  000000                  MID21:  .WORD   0
002324  000000                  MID31:  .WORD   0
002326  000000                  MID41:  .WORD   0
002330  000000                  MID51:  .WORD   0
002332  000000                  TQU10:  .WORD   0       ;3RD QUARTER TRK HEAD 0
002334  000000                  TQU20:  .WORD   0
002336  000000                  TQU30:  .WORD   0
002340  000000                  TQU40:  .WORD   0
002342  000000                  TQU50:  .WORD   0
002344  000000                  TQU11:  .WORD   0       ;3RD QUARTER TRK HEAD 1
002346  000000                  TQU21:  .WORD   0
002350  000000                  TQU31:  .WORD   0
002352  000000                  TQU41:  .WORD   0
002354  000000                  TQU51:  .WORD   0
002356  000000                  INN10:  .WORD   0       ;INNER TRK HEAD 0
002360  000000                  INN20:  .WORD   0
002362  000000                  INN30:  .WORD   0
002364  000000                  INN40:  .WORD   0
002366  000000                  INN50:  .WORD   0
002370  000000                  INN11:  .WORD   0       ;INNER TRK HEAD 1
002372  000000                  INN21:  .WORD   0
```

```
GLOBAL DATA SECTION

     58 002374  000000                 INN31:  .WORD   0
     59 002376  000000                 INN41:  .WORD   0
     60 002400  000000                 INN51:  .WORD   0
     61                                 .EVEN
     62
     63                                 ;SECTOR LIST FOR LAST DRIVE WRITTEN
     64                                 ;MAP OF 16 SECTOR DRIVE BITS
     65
     66 002402                          SECLST: .BLKW   16.
     67
     68                                 ;BUFFER TABLE FOR 24 X 5 MATRIX USED FOR ADJACENT CYLINDER TESTING.
     69
     70 002442                          SECBUF: .BLKW   5*24.
     71
     72                                 ;LIST OF TRACKS USED TO OVERWRITE TEST.
     73                                 ;FIRST FIVE ARE CYLINDER ADDRESSES OF TOP SURFACE.
     74                                 ;LAST FIVE ARE CYLINDER ADDRESSES OF BOTTOM SURFACE.
     75
     76 003022  002242                  OVWTRK: OUT30
     77 003024  002266                          OQU30
     78 003026  002312                          MID30
     79 003030  002336                          TQU30
     80 003032  002362                          INN30
     81 003034  002254                          OUT31
     82 003036  002300                          OQU31
     83 003040  002324                          MID31
     84 003042  002350                          TQU31
     85 003044  002374                          INN31
     86
     87 003046  152525                  PATLST: .WORD   152525
     88 003050  133333                          .WORD   133333
     89 003052  066666                          .WORD   066666
     90 003054  155555                          .WORD   155555
     91
```

```
003056	000000		TEM:	.WORD	0
003060	000000		T.DRIVE: .WORD	0
003062	000000		FOWR:	.WORD	0
003064	000000		FADJ:	.WORD	0
003066	000000		TEMP:	.WORD	0
003070	000000		LSTCLR: .WORD	0		;LAST CONTROLLER
003072	000000		REASON: .WORD	0		;DRIVE ERROR REASON
003074	000000		ERFLG:	.WORD	0		;ERROR FLAG
003076	000000		STFLG:	.WORD	0		;PROGRAM START UP FLAG
003100	000000		ADJLOC: .WORD	0		;TRACK INDEX FOR ADJ. CYL TEST
003102	000000		ADJFLG: .WORD	0		;FLAG FOR ADJ. STORE OR RETRIEVE
003104	000000		ADJDIR: .WORD	0		;ADJACENT SEEK DIRECTION
003106	000000		DRSTAT: .WORD	0
003110	000000		HSFLG:	.WORD	0
003112	000000		OSECT:	.WORD	0
003114	000000		HEAD01: .WORD	0		;SURFACE FLAG
003116	000000		DIRC:	.WORD	0		;DIRECTION OF SEEK
003120	000000		SURF:	.WORD	0
003122	000000		CYL:	.WORD	0
003124	000000		REVSK:	.WORD	0		;REVERSE SEEK
003126	000000		FORSK:	.WORD	0		;FORWARD SEEK
003130	000000		UUT:	.WORD	0		;UNIT UNDER TEST
003132	000000		SECT:	.WORD	0		;SECTOR
003134	000000		LSTDRV: .WORD	0		;LAST DRIVE
003136	000000		GDATA:	.WORD	0		;GOOD DATA
003140	000000		BDATA:	.WORD	0		;BAD DATA
003142	000000		WCOUNT: .WORD	0		;WORD COUNT
003144	000000		SECWRD: .WORD	0		;SECTOR WORD
003146	000000		OFFSET: .WORD	0		;INCREMENT
003150	000000		LSTTRK: .WORD	0		;LAST TRACK OF SEARCH
003152	000000		FRTTRK: .WORD	0		;FIRST TRACK OF SEARCH
003154	000000		PRSTRK: .WORD	0		;PRESENT TRACK
003156	000000		SURFACE: .WORD	0		;SURFACE
003160	000000		TRKFND: .WORD	0		;TRACK FOUND
003162	000000		TRKCNT: .WORD	0		;TRACK COUNT
003164	000000		E.CS:	.WORD	0		;IMAGE OF CSR
003166	000000		E.BA:	.WORD	0		;IMAGE OF BUS ADDRESS
003170	000000		E.DA:	.WORD	0		;IMAGE OF DISK ADDRESS
003172	000000		E.MP:	.WORD	0		;IMAGE OF MULTI-PURPOSE WORD 1
003174	000000		E.MP1:	.WORD	0		;  "   "    "      "      "  2
003176	000000		E.MP2:	.WORD	0		;  "   "    "      "      "  3
003200	000000		BCS:	.WORD	0		;COMMAND LOADED
003202	000000		BBA:	.WORD	0		;BUS ADDRESS LOADED
003204	000000		BDA:	.WORD	0		;DISK ADDRESS LOADED
003206	000000		BMP:	.WORD	0		;WORD COUNT LOADED
003210	000000		SERNM1: .WORD	0		;SERIAL NUMBER OF CATRIDGE
003212	000000		SERNM2: .WORD	0		;  "     "     "     "
003214	000000		ADJTRK: .WORD	0		;INSIDE/OUTSIDE FLAG
003216	000000		ADJUUT: .WORD	0		;UUT FOR "ADJCYL"
003220	000000		ADJLC2: .WORD	0		;TEMP LOC FOR "ADJCYL"
003222	000000		ADJLC3: .WORD	0		;  "   "   "    "
003224	000000		ADJLC4: .WORD	0		;  "   "   "    "
003226	000000		STSEC1: .WORD	0		;SECTORS TO WRITE "ADJCYL"
003230	000000		STSEC:	.WORD	0		;  "     "    "     "
003232				BUF:	.BLKW	3072.		;BUFFER FOR 24 SECTOR READS
017232	000000		XDELAY: .WORD	0		;DELAY FOR WAIT MICRO-SECOND MACRO
```

GLOBAL DATA SECTION

```
     58 017234  000000                YDELAY: .WORD   0               ;DELAY FOR WAIT MILLI-SECOND MACRO
     59 017236  000000                OBUFF:  .WORD   0               ;RESPONSE BUFFER
     60
     61
     62 017240                        DRBUF:                          ;DRIVE INFORMATION BUFFERS
     63
     67
     68         000004                .REPT   4.
     75
        017240  000000                        CSR                     ;CONTROLLER ADDRESS
        017242  000002                        VEC                     ;VECTOR
        017244  000004                        DSB                     ;DRIVE SELECT BITS
        017246  000006                        PAT                     ;PATTERN UNIQUE TO DRIVE

        017250  000000                        CSR                     ;CONTROLLER ADDRESS
        017252  000002                        VEC                     ;VECTOR
        017254  000004                        DSB                     ;DRIVE SELECT BITS
        017256  000006                        PAT                     ;PATTERN UNIQUE TO DRIVE

        017260  000000                        CSR                     ;CONTROLLER ADDRESS
        017262  000002                        VEC                     ;VECTOR
        017264  000004                        DSB                     ;DRIVE SELECT BITS
        017266  000006                        PAT                     ;PATTERN UNIQUE TO DRIVE

        017270  000000                        CSR                     ;CONTROLLER ADDRESS
        017272  000002                        VEC                     ;VECTOR
        017274  000004                        DSB                     ;DRIVE SELECT BITS
        017276  000006                        PAT                     ;PATTERN UNIQUE TO DRIVE

     76
     80 017300  000000                ENDBUF: .WORD   0               ;END OF DRIVE BUFFERS
     81 017302                                ENDMOD
```

```
                                                .SBTTL GLOBAL TEXT SECTION
 017302                                         BGNMOD  GLBTXT

                                                ;GLOBAL TEXT



 017302     103     117     116     OPR001: .ASCIZ  /CONTINUE TEST?/
 017321     101     102     117     OPR002: .ASCIZ  /ABOVE CONDITIONS MET/
 017346     103     117     116     CNTTOT: .ASCIZ  /CONTROLLER TIMED OUT/
 017373     105     122     122     INITWR: .ASCIZ  /ERROR ON RECOVERING INITIAL WRITE BY FIRST DRIVE /
 017455     105     122     122     DCKER:  .ASCIZ  /ERROR ON READ/
 017473     115     111     116     FEW:    .ASCIZ  /MINIMUM OF TWO DRIVES REQUIRED/
 017532     115     101     130     MANY:   .ASCIZ  /MAXIMUM OF FOUR DRIVES ALLOWED/
 017571     124     105     123     NONE:   .ASCIZ  /TEST ABORTED - CAN'T FIND ANY GOOD SPOTS/
 017642     124     122     131     OVMES:  .ASCIZ  /TRYING TO OVERWRITE DRIVE /
 017675     124     122     131     RECMS:  .ASCIZ  /TRYING TO READ DATA WRITTEN BY DRIVE /
 017743     103     101     116     ERRFND: .ASCIZ  /CAN'T FIND FIVE ADJACENT TRACKS/
 020003     117     126     105     OVWER:  .ASCIZ  /OVERWRITE ERROR/
 020023     122     105     101     RECER:  .ASCIZ  /READ RECOVERY ERROR/
 020047     105     122     122     FUNERR: .ASCIZ  /ERROR IN SEEK OPERATION/
 020077     115     111     123     SKER:   .ASCIZ  /MIS SEEK ERROR/
 020116     106     117     122     FWD:    .ASCIZ  /FORWARD/
 020126     122     105     126     REV:    .ASCIZ  /REVERSE/
 020136     105     122     122     WRIT1:  .ASCIZ  /ERROR WRITING SECTOR/
 020163     105     122     122     READ1:  .ASCIZ  /ERROR READING SECTOR/
 020210     101     104     112     ADJTXT: .ASCIZ  /ADJACENT CYLINDER TEST/
                                    .EVEN

 020240                             ENDMOD

                                    .SBTTL GLOBAL ERROR REPORT SECTION

 020240                             BGNMOD  GLBERR

 020240                             BGNMSG  ERR1

 020240                                     PRINTB  #FRM10,FRTTRK,LSTTRK,SURFACE     ;BETWEEN _ _ HEAD _
 020240  013746  003156                     MOV     SURFACE,-(SP)
 020244  013746  003150                     MOV     LSTTRK,-(SP)
 020250  013746  003152                     MOV     FRTTRK,-(SP)
 020254  012746  021545                     MOV     #FRM10,-(SP)
 020260  012746  000004                     MOV     #4,-(SP)
 020264  010600                             MOV     SP,R0
 020266  104414                             TRAP    C$PNTB
 020270  062706  000012                     ADD     #12,SP

 020274                                     ENDMSG
 020274                             L10000:
 020274  104423                             TRAP    C$MSG

 020276                             BGNMSG  ERR2
 020276                                     PRINTB  #FRM4,CSR(R4),<B,DSB+1(R4)>      ;CONTROLLER _ DRIVE _
 020276  005046                             CLR     -(SP)
 020300  156416  000005                     BISB    DSB+1(R4),(SP)
 020304  016446  000000                     MOV     CSR(R4),-(SP)
 020310  012746  021246                     MOV     #FRM4,-(SP)
```

```
                020314  012746  000003          MOV     #3,-(SP)
                020320  010600                  MOV     SP,R0
                020322  104414                  TRAP    C$PNTB
                020324  062706  000010          ADD     #10,SP
     47         020330  004737  026522          JSR     PC,REGDMP                       ;REGISTER DUMP ROUTINE
     48         020334                          ENDMSG
                020334                  L10001:
                020334  104423                  TRAP    C$MSG
     49
     50         020336                  BGNMSG  ERR3
     51         020336                          PRINTB  #FRM4,CSR(R4),<8,DSB+1(R4)>     ;CONTROLLER _ DRIVE _
                020336  005046                  CLR     -(SP)
                020340  156416  000005          BISB    DSB+1(R4),(SP)
                020344  016446  000000          MOV     CSR(R4),-(SP)
                020350  012746  021246          MOV     #FRM4,-(SP)
                020354  012746  000003          MOV     #3,-(SP)
                020360  010600                  MOV     SP,R0
                020362  104414                  TRAP    C$PNTB
                020364  062706  000010          ADD     #10,SP
     52         020370  004737  026522          JSR     PC,REGDMP                       ;REGISTER DUMP ROUTINE
     53         020374                          PRINTB  #FRM5,<SURF>,<CYL>,SECT         ;HEAD _ CYLINDER _ SECTOR _
                020374  013746  003132          MOV     SECT,-(SP)
                020400  013746  003122          MOV     CYL,-(SP)
                020404  013746  003120          MOV     SURF,-(SP)
                020410  012746  021307          MOV     #FRM5,-(SP)
                020414  012746  000004          MOV     #4,-(SP)
                020420  010600                  MOV     SP,R0
                020422  104414                  TRAP    C$PNTB
                020424  062706  000012          ADD     #12,SP
                                                                                        ;ADJACENT WRITTEN BY CONTROLLER
     54
     55         020430                          PRINTB  #FRM16,CSR(R3),<8,DSB+1(R3)>    ;_ DRIVE _
                020430  005046                  CLR     -(SP)
                020432  156316  000005          BISB    DSB+1(R3),(SP)
                020436  016346  000000          MOV     CSR(R3),-(SP)
                020442  012746  022076          MOV     #FRM16,-(SP)
                020446  012746  000003          MOV     #3,-(SP)
                020452  010600                  MOV     SP,R0
                020454  104414                  TRAP    C$PNTB
                020456  062706  000010          ADD     #10,SP
     56
     57         020462                          ENDMSG
                020462                  L10002:
                020462  104423                  TRAP    C$MSG
     58
     59         020464                  BGNMSG  ERR4
     60
     61         020464                          PRINTB  #FRM4,CSR(R4),<8,DSB+1(R4)>     ;CONTROLLER _ DRIVE _
                020464  005046                  CLR     -(SP)
                020466  156416  000005          BISB    DSB+1(R4),(SP)
                020472  016446  000000          MOV     CSR(R4),-(SP)
                020476  012746  021246          MOV     #FRM4,-(SP)
                020502  012746  000003          MOV     #3,-(SP)
                020506  010600                  MOV     SP,R0
                020510  104414                  TRAP    C$PNTB
                020512  062706  000010          ADD     #10,SP
     62         020516  004737  026522          JSR     PC,REGDMP                       ;REGISTER DUMP ROUTINE
     63         020522                  PRINTB  #FRM5,<SURF>,<CYL>,SECT                 ;HEAD _ CYLINDER _ SECTOR _
```

```
                020522  013746  003132          MOV     SECT,-(SP)
                020526  013746  003122          MOV     CYL,-(SP)
                020532  013746  003120          MOV     SURF,-(SP)
                020536  012746  021307          MOV     #FRM5,-(SP)
                020542  012746  000004          MOV     #4,-(SP)
                020546  010600                  MOV     SP,R0
                020550  104414                  TRAP    C$PNTB
                020552  062706  000012          ADD     #12,SP
             64 020556                          PRINTB  #FRM6,REASON,LSTDRV,LSTCLR,LSTDRV
                020556  013746  003134          MOV     LSTDRV,-(SP)
                020562  013746  003070          MOV     LSTCLR,-(SP)
                020566  013746  003134          MOV     LSTDRV,-(SP)
                020572  013746  003072          MOV     REASON,-(SP)
                020576  012746  021356          MOV     #FRM6,-(SP)
                020602  012746  000005          MOV     #5,-(SP)
                020606  010600                  MOV     SP,R0
                020610  104414                  TRAP    C$PNTB
                020612  062706  000014          ADD     #14,SP
             65 020616                          PRINTB  #FRM7,DIRC                      ;SEEK DIRECTION
                020616  013746  003116          MOV     DIRC,-(SP)
                020622  012746  021377          MOV     #FRM7,-(SP)
                020626  012746  000002          MOV     #2,-(SP)
                020632  010600                  MOV     SP,R0
                020634  104414                  TRAP    C$PNTB
                020636  062706  000006          ADD     #6,SP
             66
             67 020642                          ENDMSG
                020642                  L10003:
                020642  104423                  TRAP    C$MSG
             68
             69 020644                  BGNMSG  ERR5
             70 020644                          PRINTB  #FRM4,CSR(R4),<B,DSB+1(R4)>     ;CONTROLLER _ DRIVE _
                020644  005046                  CLR     -(SP)
                020646  156416  000005          BISB    DSB+1(R4),(SP)
                020652  016446  000000          MOV     CSR(R4),-(SP)
                020656  012746  021246          MOV     #FRM4,-(SP)
                020662  012746  000003          MOV     #3,-(SP)
                020666  010600                  MOV     SP,R0
                020670  104414                  TRAP    C$PNTB
                020672  062706  000010          ADD     #10,SP
             71 020676  004737  026522          JSR     PC,REGDMP
             72 020702                          ENDMSG
                020702                  L10004:
                020702  104423                  TRAP    C$MSG
             73
             74 020704                  BGNMSG  ERR6
             75 020704                          PRINTB  #FRM4,CSR(R4),<B,DSB+1(R4)>
                020704  005046                  CLR     -(SP)
                020706  156416  000005          BISB    DSB+1(R4),(SP)
                020712  016446  000000          MOV     CSR(R4),-(SP)
                020716  012746  021246          MOV     #FRM4,-(SP)
                020722  012746  000003          MOV     #3,-(SP)
                020726  010600                  MOV     SP,R0
                020730  104414                  TRAP    C$PNTB
                020732  062706  000010          ADD     #10,SP
             76 020736  004737  026522          JSR     PC,REGDMP
             77 020742                          PRINTB  #FRM17,R1,E.MP
```

```
GLOBAL ERROR REPORT SECTION

        020742  013746  003172                  MOV     E.MP,-(SP)
        020746  010146                          MOV     R1,-(SP)
        020750  012746  022163                  MOV     #FRM17,-(SP)
        020754  012746  000003                  MOV     #3,-(SP)
        020760  010600                          MOV     SP,R0
        020762  104414                          TRAP    C$PNTB
        020764  062706  000010                  ADD     #10,SP
     78 020770                                  ENDMSG
        020770                          L10005:
        020770  104423                          TRAP    C$MSG
     79
     80
     81
     82                                 ;FORMAT STATMENTS
     83
     87
     88 020772      045     116     045 FRM1:   .ASCIZ  /%N%AUNLOAD DRIVE %01%A ON CONTROLLER %06%A AND REMOVE PACK%N/
     89 021067      045     116     045 FRM2:   .ASCIZ  /%N%APLACE PACK IN DRIVE %01%A ON CONTROLLER %06%A AND LOAD IT%N/
     90 021167      045     116     045 FRM3:   .ASCIZ  !%N%AWRONG PACK  # IS %05%05%A # S/B %05%05%N%N!
     91 021246      045     101     103 FRM4:   .ASCIZ  /%ACONTROLLER: %06%A DRIVE: %01%N/
     92 021307      045     101     110 FRM5:   .ASCIZ  /%AHEAD: %01%A CYL: %Z3%A SECTOR: %Z2%N/
     93 021356      045     124     045 FRM6:   .ASCIZ  /%T%01%A ON %06%N/
     94 021377      045     101     123 FRM7:   .ASCIZ  /%ASEEK DIRECTION: %T%N%ADATA:%N/
     95 021437      045     101     127 FRM8:   .ASCIZ  !%AWORD: %Z3%A S/B: %06%A WAS: %06%N!
     96 021503      045     104     063 FRM9:   .ASCIZ  /%D3%A WORDS BAD OUT OF 128 READ%N/
     97 021545      045     101     102 FRM10:  .ASCIZ  /%ABETWEEN %Z3%A - %Z3%A HEAD: %01%N/
     98 021611      045     116     045 FRM11:  .ASCIZ  /%N%APWR FAIL NOT SUPPORTED%N/
     99 021646      045     101     102 FRM12:  .ASCIZ  /%ABEFORE CS: %06%A BA: %06%A DA: %06%A MP: %06/
    100 021725      045     116     045 FRM13:  .ASCIZ  /%N%AAFTER  CS: %06%A BA: %06%A DA: %06%A MP: %06%N/
    101 022010      045     116     045 FRM14:  .ASCIZ  /%N%A DRIVE STATUS: %06/
    102 022037      045     116     045 FRM15:  .ASCIZ  /%N%ACAN'T FIND BAD SECTOR FILE/
    103 022076      045     101     101 FRM16:  .ASCIZ  /%AADJACENT WRITTEN BY CONTROLLER: %06%A DRIVE: %01%N/
    104 022163      045     101     105 FRM17:  .ASCIZ  /%AEXP'D: %06%A REC'D: %06%N/
    105 022217      045     116     045 FRM18:  .ASCIZ  /%N%AUNLOAD AND WRITE ENABLE ALL DRIVES TO BE USED%N/
    106 022303      045     116     045 FRM19:  .ASCIZ  /%N%ADRIVE TYPE IS DIFFERNT.%N/
    107 022341      045     116     045 FRM20:  .ASCIZ  /%N%ADRIVE NUMBER PREVIOUSLY SPECIFIED.%N/
    108 022412      045     116     045 ENDPAS: .ASCIZ  /%N%A END OF TEST%N%N/
    109
    113                                 .EVEN
    114
    115 022440                          ENDMOD
    116
```

```
      1
      2                                 ;LOAD PROTECTION TABLE
      3
      4 022440                          BGNPROT
      5
      6 022440  000000                          .WORD    0              ;OFFSET OF CSR IN P-TABLE
      7 022442  177777                          .WORD   -1              ;NOT A MASS-BUS DRIVE
      8 022444  000006                          .WORD    6              ;OFFSET OF DRIVE IN P-TABLE
      9
     10 022446                          ENDPROT
     11
     12
     13 022446                          BGNMOD  HPTCODE
     14 022446                          BGNHW
        022446  000005                          .WORD   L10007-L$HW/2
     15 022450  174400                          .WORD   174400          ;BASE ADDRESS DEFAULT
     16 022452  000160                          .WORD   160             ;VECTOR DEFAULT
     17 022454  000240                          .WORD   240             ;PRIORITY DEFAULT
     18 022456  000001                          .WORD   1               ;RL01 OR RL02 (RL01=1)
     19 022460  000000                          .WORD   0               ;DRIVE NUMBER DEFAULT
     20 022462                          ENDHW
        022462                          L10007:
     21
     22 022462                          ENDMOD
     23
     24
     25 022462                          BGNMOD  DSPCODE
     26
     27 022462                          DISPATCH        1
        022462  000001                          .WORD   1
        022464  032706                          .WORD   T1
     28
     29 022466                          ENDMOD
     30
     31
```

```
                                .SBTTL INITIALIZATION SECTION

022466                          BGNMOD  INITCODE

022466                                  BGNINIT

022466                                  SETPRI  #340
022466  012700  000340                  MOV     #340,R0
022472  104441                          TRAP    C$SPRI

022474  023727  002012  000002          CMP     L$UNIT,#2               ;MORE THAN TWO
022502  002006                          BGE     90$                     ;YES, OKAY

022504                                  ERRSF   19.,FEW                 ;MININUM OFF TWO DRIVE REQUIRED
022504  104454                          TRAP    C$ERSF
022506  000023                          .WORD   19
022510  017473                          .WORD   FEW
022512  000000                          .WORD   0
022514  000137  024266                  JMP     CMPENA                  ;CLEAN CODE WHEN < 2 DRIVES

022520  023727  002012  000004  90$:    CMP     L$UNIT,#4               ;MORE THAN FOUR
022526  003406                          BLE     91$                     ;NO, OKAY

022530                                  ERRSF   20.,MANY                ;MAXIMUM OF FOUR DRIVES ALLOWED
022530  104454                          TRAP    C$ERSF
022532  000024                          .WORD   20
022534  017532                          .WORD   MANY
022536  000000                          .WORD   0
022540  000137  024266                  JMP     CMPENA                  ;CLEAN CODE WHEN > 4 DRIVES

022544  013737  002012  003130  91$:    MOV     L$UNIT,UUT              ;GET NUMBER OF UNITS
022552  005001                          CLR     R1                      ;INIT P-TABLE
022554  012704  017240                  MOV     #DRBUF,R4               ;SET UP DRIVE BUFFER
022560  012702  003046                  MOV     #PATLST,R2              ;GET LIST OF PATTERNS
022564  005737  003130          1$:     TST     UUT                     ;ANY    P-TABLES LEFT?
022570  001513                          BEQ     END                     ;NO,GO TO END
022572                                  GPHARD  R1,R0                   ;GET A P-TABLE
022572  010100                          MOV     R1,R0
022574  104442                          TRAP    C$GPHRD
022576  012064  000000                  MOV     (R0)+,CSR(R4)           ;GET CSR
022602  012064  000002                  MOV     (R0)+,VEC(R4)           ;GET VECTOR
022606  012064  000004                  MOV     (R0)+,PRIOR(R4)         ;GET BREAK LEVEL
022612  012037  003060                  MOV     (R0)+,T.DRIVE           ;RL01/2 TYPE ... RL01=1
022616  011064  000004                  MOV     (R0),DSB(R4)            ;GET DRIVE
022622  011264  000006                  MOV     (R2),PAT(R4)
022626  005722                          TST     (R2)+

                                        ;TEST FOR DRIVES OF SAME TYPE AND NO REPEATED DRIVE NUMBERS

022630  023737  002012  003130          CMP     L$UNIT,UUT              ;SKIP TEST FOR FIRST DRIVE
022636  001462                          BEQ     6$

022640                                  GPHARD  #0,R5                   ;BASE ADDRESS OF FIRST P TABLE
022640  012700  000000                  MOV     #0,R0
022644  104442                          TRAP    C$GPHRD
022646  010005                          MOV     R0,R5

```

```
INITIALIZATION SECTION

     43 022650  023765  003060  000006          CMP     T.DRIVE,TYPDR(R5)       ;CHECK DRIVE TYPE
     44 022656  001423                          BEQ     4$
     45 022660                                  PRINTF  #FRM19                  ;PROMPT - DRIVE TYPE DIFFERNT ...
        022660  012746  022303                  MOV     #FRM19,-(SP)
        022664  012746  000001                  MOV     #1,-(SP)
        022670  010600                          MOV     SP,R0
        022672  104417                          TRAP    C$PNTF
        022674  062706  000004                  ADD     #4,SP
     46
     47 022700                                  GMANIL  OPR001,OBUFF,1,YES      ;PROMPY - CONTINUE TEST
        022700  104443                          TRAP    C$GMAN
        022702  000404                          BR      10000$
        022704  017236                          .WORD   OBUFF
        022706  000130                          .WORD   T$CODE
        022710  017302                          .WORD   OPR001
        022712  000001                          .WORD   1
        022714                          10000$:
     48 022714  005737  017236                  TST     OBUFF
     49 022720  001002                          BNE     4$
     50 022722  000137  024266                  JMP     CMPENA                  ;RETURN TO SUPERVISOR
     51
     52 022726  026465  000004  000010  4$:     CMP     DSB(R4),DRBT(R5)        ;CHECK DRIVE NUMBER
     53 022734  001023                          BNE     6$
     54 022736                                  PRINTF  #FRM20                  ;PROMPT - DRIVE NUMBER ..
        022736  012746  022341                  MOV     #FRM20,-(SP)
        022742  012746  000001                  MOV     #1,-(SP)
        022746  010600                          MOV     SP,R0
        022750  104417                          TRAP    C$PNTF
        022752  062706  000004                  ADD     #4,SP
     55
     56 022756                                  GMANIL  OPR001,OBUFF,1,YES      ;PROMPT - CONTINUE TEST
        022756  104443                          TRAP    C$GMAN
        022760  000404                          BR      10001$
        022762  017236                          .WORD   OBUFF
        022764  000130                          .WORD   T$CODE
        022766  017302                          .WORD   OPR001
        022770  000001                          .WORD   1
        022772                          10001$:
     57 022772  005737  017236                  TST     OBUFF
     58 022776  001002                          BNE     6$
     59 023000  000137  024266                  JMP     CMPENA                  ;RETURN TO SUPERVISOR
     60
     61 023004  005201                  6$:     INC     R1                      ;NEXT P TABLE
     62 023006  005337  003130                  DEC     UUT                     ;NEXT DRIVE
     63 023012  062704  000010                  ADD     #PAT+2,R4
     64 023016  000662                          BR      1$
     65 023020  013737  002012  003130  END:    MOV     L$UNIT,UUT
     66 023026  012704  017240                  MOV     #DRBUF,R4               ;GET BEGINNING OF BUFFER
     67 023032  005037  003064                  CLR     FADJ                    ;CLEAR ADJ. TEST FLAG
     68 023036  005037  003062                  CLR     FOWR                    ;CLEAR OVERWRITE FLAG
     69 023042                                  READEF  #EF.PWR
        023042  012700  000034                  MOV     #EF.PWR,R0
        023046  104447                          TRAP    C$REFG
     70 023050                                  BNCOMPLETE      SETUP
        023050  103010                          BCC     SETUP
     71 023052                                  PRINTF  #FRM11                  ;PROMPT - PWR FAIL NOT SUPPORTED
        023052  012746  021611                  MOV     #FRM11,-(SP)
```

INITIALIZATION SECTION

```
        023056  012746  000001                  MOV     #1,-(SP)
        023062  010600                          MOV     SP,R0
        023064  104417                          TRAP    C$PNTF
        023066  062706  000004                  ADD     #4,SP
     72
     73
     74                                 ;INITIALIZE ROUTINE
     75                                 ;WE ATTEMPT TO LOCATE 5 PERFECT ADJACENT TRACKS AT 5 SPOTS
     76                                 ;ACROSS THE PACK.
     77                                 ;THE 5 SPOTS ARE: (EACH SURFACE)
     78                                 ;
     79                                 ;OUTER - TRACK 0 - 16 (BOTH RL01 & RL02)
     80                                 ;INNER - TRACK 238 - 254 (RL01) OR 494 - 510 (RL02)
     81                                 ;MIDDLE - TRACK 120 - 136 (RL01) OR 248 - 264 (RL02)
     82                                 ;ONE QUARTER - TRACK 56 - 72 (RL01) OR 120 - 136 (RL02)
     83                                 ;THREE QUARTER - TRACK 184 - 200 (RL01) OR 376 - 392 (RL02)
     84                                 ;
     85                                 ;IF WE FIND ANY BAD SPOTS, WE WILL REPORT SO.....
     86
     87
     88 023072  005237  003076          SETUP:  INC     STFLG                   ;INDICATE A START COMMAND
     89 023076  012737  177777  003210          MOV     #-1,SERNM1
     90 023104  012737  177777  003212          MOV     #-1,SERNM2
     91 023112                          1$:     PRINTF  #FRM18                  ;PROMPT - UNLOAD DRIVES TO BE USED
        023112  012746  022217                  MOV     #FRM18,-(SP)
        023116  012746  000001                  MOV     #1,-(SP)
        023122  010600                          MOV     SP,R0
        023124  104417                          TRAP    C$PNTF
        023126  062706  000004                  ADD     #4,SP
     92 023132                                  GMANIL  OPR002,OBUFF,1, NO      ;PROMPT - ABOVE CONDITIONS MET
        023132  104443                          TRAP    C$GMAN
        023134  000404                          BR      10002$
        023136  017236                          .WORD   OBUFF
        023140  000120                          .WORD   T$CODE
        023142  017321                          .WORD   OPR002
        023144  000001                          .WORD   1
        023146                          10002$:
     93 023146  005737  017236                  TST     OBUFF                   ;NO - ASK AGAIN
     94 023152  001757                          BEQ     1$
     95
     96 023154  004537  032300                  JSR     R5,LOAD                 ;TELL OPERATOR TO LOAD
     97 023160  004537  031526                  JSR     R5,SERNUM               ;GET SERIAL NUMBER
     98 023164  004537  031002                  JSR     R5,MERGE                ;MERGE BAD SECTOR FILES
     99 023170  012701  002236                  MOV     #OUT10,R1               ;INITIALIZE ALL TRACKS
    100 023174  012700  000062                  MOV     #50.,R0
    101 023200  012721  177777          3$:     MOV     #177777,(R1)+
    102 023204  005300                          DEC     R0
    103 023206  001374                          BNE     3$
    104
    105 023210  004537  031230                  JSR     R5,FNDTRK               ;TRY TO FIND FIVE TRACKS
    106 023214  000001                          1                               ;INWARD SEARCH
    107 023216  000000                          0                               ;TOP SURFACE
    108
    109 023220  000000  000020                  .WORD   0,16.
    110 023224  000000  000020                  .WORD   0,16.
    111
    112 023230  005737  003160                  TST     TRKFND                  ;WAS SEARCH SUCCESSFUL????
```

```
113 023234  001005                          BNE     5$                      ;YES
114
115 023236                                   ERRHRD  10.,ERRFND,ERR1         ;CAN'T FIND 5 ADJACENT TRACKS
    023236  104456                          TRAP    C$ERHRD
    023240  000012                          .WORD   10
    023242  017743                          .WORD   ERRFND
    023244  020240                          .WORD   ERR1
116 023246  000404                          BR      7$
117
118 023250  012700  002236          5$:     MOV     #OUT10,R0               ;STORE AWAY TRACKS FOUND
119 023254  004537  031472                  JSR     R5,FIXCYL
120
121 023260  004537  031230          7$:     JSR     R5,FNDTRK               ;TRY TO FIND FIVE TRACKS
122 023264  000001                          1                               ;INWARD SEARCH
123 023266  000001                          1                               ;BOTTOM SURFACE
124 023270  000000  000020                  .WORD   0,16.
125 023274  000000  000020                  .WORD   0,16.
126
127 023300  005737  003160                  TST     TRKFND                  ;WAS    SEARCH  SUCCESSFUL????
128 023304  001005                          BNE     9$                      ;YES
129
130 023306                                   ERRHRD  10.,ERRFND,ERR1         ;CAN'T FIND 5 ADJACENT TRACKS
    023306  104456                          TRAP    C$ERHRD
    023310  000012                          .WORD   10
    023312  017743                          .WORD   ERRFND
    023314  020240                          .WORD   ERR1
131 023316  000404                          BR      10$
132
133 023320  012700  002250          9$:     MOV     #OUT11,R0               ;STORE TRACKS AWAY
134 023324  004537  031472                  JSR     R5,FIXCYL
135 023330  004537  031230          10$:    JSR     R5,FNDTRK               ;FIND NEXT 5 TRACK
136 023334  177777                          -1                              ;OUTWARD SEARCH
137 023336  000000                          0                               ;TOP SURFACE
138 023340  000376  000356                  .WORD   254.,238.               ;TRACK RANGE
139 023344  000776  000756                  .WORD   510.,494.
140
141 023350  005737  003160                  TST     TRKFND                  ;WAS SEARCH SUCCESSFUL?
142 023354  001005                          BNE     12$                     ;YES
143
144 023356                                   ERRHRD  10.,ERRFND,ERR1         ;CAN'T FIND 5 ADJACENT TRACKS
    023356  104456                          TRAP    C$ERHRD
    023360  000012                          .WORD   10
    023362  017743                          .WORD   ERRFND
    023364  020240                          .WORD   ERR1
145 023366  000404                          BR      14$                     ;SKIP
146
147 023370  012700  002356          12$:    MOV     #INN10,R0               ;STORE AWAY TRACKS FOUND
148 023374  004537  031472                  JSR     R5,FIXCYL
149
150 023400  004537  031230          14$:    JSR     R5,FNDTRK               ;NEXT SET
151 023404  177777                          -1                              ;OUTWARD SEARCH
152 023406  000001                          1                               ;BOTTOM SURFACE
153 023410  000376  000356                  .WORD   254.,238.
154 023414  000776  000756                  .WORD   510.,494.
155
156 023420  005737  003160                  TST     TRKFND                  ;SEARCH SUCCESSFUL?
157 023424  001005                          BNE     16$                     ;YES
```

```
                                                ERRHRD  10.,ERRFND,ERR1          ;CAN'T FIND 5 ADJACENT TRACKS
023426  104456                                  TRAP    C$ERHRD
023430  000012                                  .WORD   10
023432  017743                                  .WORD   ERRFND
023434  020240                                  .WORD   ERR1
023436  000404                                  BR      18$

023440  012700  002370          16$:            MOV     #INN11,R0               ;STORE AWAY TRACKS FOUND
023444  004537  031472                          JSR     R5,FIXCYL

023450  004537  031230          18$:            JSR     R5,FNDTRK               ;NEXT SET
023454  000001                                  1                               ;INWARD SEARCH
023456  000000                                  0                               ;TOP SURFACE
023460  000176  000210                          .WORD   126.,136.               ;TRACK RANGE
023464  000376  000410                          .WORD   254.,264.

023470  005737  003160                          TST     TRKFND                  ;DID WE FIND A SET
023474  001020                                  BNE     20$                     ;YES

023476  004537  031230                          JSR     R5,FNDTRK               ;NEXT SET (OTHER SIDE)
023502  177777                                  -1                              ;OUTWARD SEARCH
023504  000000                                  0                               ;TOP SURFACE
023506  000202  000170                          .WORD   130.,120.               ;TRACK RANGE
023512  000402  000370                          .WORD   258.,248.
023516  005737  003160                          TST     TRKFND                  ;DID WE FIND A SET
023522  001005                                  BNE     20$                     ;YES

023524                                          ERRHRD  10.,ERRFND,ERR1          ;CAN'T FIND 5 ADJACENT TRACKS
023524  104456                                  TRAP    C$ERHRD
023526  000012                                  .WORD   10
023530  017743                                  .WORD   ERRFND
023532  020240                                  .WORD   ERR1
023534  000404                                  BR      22$

023536  012700  002306          20$:            MOV     #MID10,R0               ;STORE AWAY
023542  004537  031472                          JSR     R5,FIXCYL
023546  004537  031230          22$:            JSR     R5,FNDTRK               ;NEXT SET
023552  000001                                  1                               ;INWARD SEARCH
023554  000001                                  1                               ;BOTTOM SURFACE
023556  000176  000210                          .WORD   126.,136.               ;RANGE
023562  000376  000410                          .WORD   254.,264.

023566  005737  003160                          TST     TRKFND                  ;SUCCESS?
023572  001020                                  BNE     24$                     ;YES

023574  004537  031230                          JSR     R5,FNDTRK               ;LOOK THE OTHER SIDE
023600  177777                                  -1                              ;OUTWARD
023602  000001                                  1                               ;BOTTOM SURFACE
023604  000202  000170                          .WORD   130.,120.
023610  000402  000370                          .WORD   258.,248.

023614  005737  003160                          TST     TRKFND                  ;SUCCESS?
023620  001005                                  BNE     24$                     ;YES

023622                                          ERRHRD  10.,ERRFND,ERR1          ;CAN'T FIND 5 ADJACENT TRACKS
023622  104456                                  TRAP    C$ERHRD
```

INITIALIZATION SECTION

```
        023624  000012                          .WORD   10
        023626  017743                          .WORD   ERRFND
        023630  020240                          .WORD   ERR1
    206 023632  000404                          BR      26$
    207
    208 023634  012700  002320          24$:    MOV     #MID11,R0           ;STORE AWAY THE TRACKS FOUND
    209 023640  004537  031472                  JSR     R5,FIXCYL
    210
    211 023644  004537  031230          26$:    JSR     R5,FNDTRK           ;NEXT SET
    212 023650  000001                          1                           ;INWARD
    213 023652  000000                          0                           ;TOP SURFACE
    214 023654  000076  000110                  .WORD   62.,72.             ;RANGE
    215 023660  000176  000210                  .WORD   126.,136.
    216
    217 023664  005737  003160                  TST     TRKFND              ;SUCCESS?
    218 023670  001020                          BNE     28$                 ;YES
    219
    220 023672  004537  031230                  JSR     R5,FNDTRK           ;LOOK OTHER SIDE
    221 023676  177777                          -1                          ;OUTWARD
    222 023700  000000                          0                           ;TOP SURFACE
    223 023702  000102  000070                  .WORD   66.,56.             ;RANGE
    224 023706  000202  000170                  .WORD   130.,120.
    225
    226 023712  005737  003160                  TST     TRKFND              ;SUCCESS?
    227 023716  001005                          BNE     28$                 ;YES
    228
    229 023720                                  ERRHRD  10.,ERRFND,ERR1     ;CAN'T FIND 5 ADJACENT TRACKS
        023720  104456                          TRAP    C$ERHRD
        023722  000012                          .WORD   10
        023724  017743                          .WORD   ERRFND
        023726  020240                          .WORD   ERR1
    230 023730  000404                          BR      30$
    231
    232 023732  012700  002262          28$:    MOV     #OOU10,R0           ;STORE AWAY NEXT SET
    233 023736  004537  031472                  JSR     R5,FIXCYL
    234 023742  004537  031230          30$:    JSR     R5,FNDTRK           ;LOOK FOR NEXT SET
    235 023746  000001                          1                           ;INWARD
    236 023750  000001                          1                           ;BOTTOM
    237 023752  000076  000110                  .WORD   62.,72.             ;RANGE
    238 023756  000176  000210                  .WORD   126.,136.
    239
    240 023762  005737  003160                  TST     TRKFND              ;SUCCESS?
    241 023766  001020                          BNE     32$                 ;YES
    242
    243 023770  004537  031230                  JSR     R5,FNDTRK           ;LOOK FOR ANOTHER SET
    244 023774  177777                          -1                          ;OUTWARD
    245 023776  000001                          1                           ;BOTTOM
    246 024000  000102  000070                  .WORD   66.,56.             ;RANGE
    247 024004  000202  000170                  .WORD   130.,120.
    248
    249 024010  005737  003160                  TST     TRKFND              ;SUCCESS?
    250 024014  001005                          BNE     32$                 ;YES
    251
    252 024016                                  ERRHRD  10.,ERRFND,ERR1     ;CAN'T FIND 5 ADJACENT TRACKS
        024016  104456                          TRAP    C$ERHRD
        024020  000012                          .WORD   10
        024022  017743                          .WORD   ERRFND
```

```
                024024  020240                          .WORD   ERR1
    253         024026  000404                          BR      34$
    254
    255         024030  012700  002274          32$:    MOV     #OOU11,R0               ;STORE AWAY TRACKS
    256         024034  004537  031472                  JSR     R5,FIXCYL
    257
    258         024040  004537  031230          34$:    JSR     R5,FNDTRK               ;NEXT SET OF TRACKS
    259         024044  000001                          1                               ;INWARD
    260         024046  000000                          0                               ;TOP SURFACE
    261         024050  000276  000310                  .WORD   190.,200.               ;RANGE
    262         024054  000576  000610                  .WORD   382.,392.
    263
    264         024060  005737  003160                  TST     TRKFND                  ;SUCCESS?
    265         024064  001020                          BNE     36$                     ;YES
    266
    267         024066  004537  031230                  JSR     R5,FNDTRK               ;LOOK OTHER SIDE
    268         024072  177777                          -1                              ;OUTWARD SEARCH
    269         024074  000000                          0                               ;TOP
    270         024076  000302  000270                  .WORD   194.,184.
    271         024102  000602  000570                  .WORD   386.,376.
    272
    273         024106  005737  003160                  TST     TRKFND                  ;SUCCESS
    274         024112  001005                          BNE     36$                     ;YES
    275
    276         024114                                  ERRHRD  10.,ERRFND,ERR1         ;CAN'T FIND 5 ADJACENT TRACKS
                024114  104456                          TRAP    C$ERHRD
                024116  000012                          .WORD   10
                024120  017743                          .WORD   ERRFND
                024122  020240                          .WORD   ERR1
    277         024124  000404                          BR      38$
    278
    279         024126  012700  002332          36$:    MOV     #TQU10,R0               ;STORE TRACKS AWAY
    280         024132  004537  031472                  JSR     R5,FIXCYL
    281         024136  004537  031230          38$:    JSR     R5,FNDTRK               ;NEXT SET
    282         024142  000001                          1                               ;INWARD
    283         024144  000001                          1                               ;BOTTOM SURFACE
    284         024146  000276  000310                  .WORD   190.,200.               ;RANGE
    285         024152  000576  000610                  .WORD   382.,392.
    286
    287         024156  005737  003160                  TST     TRKFND                  ;SUCCESS?
    288         024162  001020                          BNE     40$                     ;YES
    289
    290         024164  004537  031230                  JSR     R5,FNDTRK               ;OTHER SET
    291         024170  177777                          -1                              ;OUTWARD
    292         024172  000001                          1                               ;BOTTOM SURFACE
    293         024174  000302  000270                  .WORD   194.,184.               ;RANGE
    294         024200  000602  000570                  .WORD   386.,376.
    295
    296         024204  005737  003160                  TST     TRKFND                  ;SUCCESS
    297         024210  001005                          BNE     40$                     ;YES
    298
    299         024212                                  ERRHRD  10.,ERRFND,ERR1         ;CZN'T FIND 5 ADJACENT TRACKS
                024212  104456                          TRAP    C$ERHRD
                024214  000012                          .WORD   10
                024216  017743                          .WORD   ERRFND
                024220  020240                          .WORD   ERR1
    300         024222  000404                          BR      42$
```

```
301
302 024224  012700  002344          40$:    MOV     #TOU11,R0               ;STORE SET AWAY
303 024230  004537  031472                  JSR     R5,FIXCYL
304
305 024234  012700  002236          42$:    MOV     #OUT10,R0               ;DID WE FIND ANY AT ALL
306 024240  012701  000062                  MOV     #50.,R1
307 024244  022720  177777          44$:    CMP     #-1,(R0)+
308 024250  001017                          BNE     EXIT
309 024252  005301                          DEC     R1
310 024254  001373                          BNE     44$
311 024256                                  ERRSF   3.,NONE
    024256  104454                          TRAP    C$ERSF
    024260  000003                          .WORD   3
    024262  017571                          .WORD   NONE
    024264  000000                          .WORD   0
312 024266  005001          CMPENA: CLR     R1
313 024270  013700  002012                  MOV     L$UNIT,R0
314 024274                          48$:    DODU    R1                      ;DO DROP UNIT
    024274  010100                          MOV     R1,R0
    024276  104451                          TRAP    C$DODU
315 024300  005201                          INC     R1
316 024302  005300                          DEC     R0
317 024304  001373                          BNE     48$
318 024306                                  DOCLN
    024306  104444                          TRAP    C$DCLN
319
320 024310                          EXIT:
321 024310                                  ENDINIT
    024310                          L10010:
    024310  104411                          TRAP    C$INIT
322 024312                                  ENDMOD
323
```

INITIALIZATION SECTION

```
      1
      2 024312                          BGNMOD  AUTOCODE                ;AUTO DROP SECTION
      3 024312                          BGNAUTO
      4
      5 024312  000240                          NOP                     ;DO NOTHING
      6
      7 024314                          ENDAUTO
        024314                          L10011:
        024314  104461                           TRAP    C$AUTO
      8 024316                          ENDMOD
      9
     10
     11 024316                          BGNMOD  CLNCODE
     12 024316                                  BGNCLN
     13
     14 024316  000240                          NOP
     15
     16 024320                                  ENDCLN
        024320                          L10012:
        024320  104412                           TRAP    C$CLEAN
     17 024322                                  ENDMOD
     18
     19 024322                          BGNMOD  DRPCODE
     20 024322                                  BGNDU
     21 024322  000240                          NOP
     22 024324                                  ENDDU
        024324                          L10013:
        024324  104453                           TRAP    C$DU
     23 024326                                  ENDMOD
     24
     25
     26
```

```
                                .SBTTL  GLOBAL SUBROUTINES SECTION

024326                          BGNMOD  GLBSUB

                                ;
                                ;TIMING ROUTINES
                                ;
                                ;CALL 1:        JSR     PC,TIME
                                ;
                                ;CALL 2:        JSR     PC,XTIME
                                ;

024326  012737  000160  002116  TIME:   MOV     #160,L$DLY              ;GET OUTER DELAY LOOP
024334  005437  017232                  NEG     XDELAY                  ;GET NEGATIVE OF MULTIPLY FACTOR
024340                                  READBUS                         ;Q-BUS?
024340  104407                          TRAP    C$RDBU
024342                                  BCOMPLETE       2$              ;BRANCH - IF YES
024342  103420                          BCS     2$
024344                          1$:     DELAY   1                       ;WAIT
024344  012727  000001                  MOV     #1,(PC)+
024350  000000                          .WORD   0
024352  013727  002116                  MOV     L$DLY,(PC)+
024356  000000                          .WORD   0
024360  005367  177772                  DEC     -6(PC)
024364  001375                          BNE     .-4
024366  005367  177756                  DEC     -22(PC)
024372  001367                          BNE     .-20
024374  005237  017232                  INC     XDELAY                  ;WAIT FACTOR EXPIRED?
024400  002761                          BLT     1$                      ;BRANCH - IF NO
024402  000422                          BR      4$                      ;EXIT
024404  012737  000150  002116  2$:     MOV     #150,L$DLY              ;GET OUTER DELAY LOOP
024412                          3$:     DELAY   1                       ;WAIT WITH RESPECT TO FONZ BUS
024412  012727  000001                  MOV     #1,(PC)+
024416  000000                          .WORD   0
024420  013727  002116                  MOV     L$DLY,(PC)+
024424  000000                          .WORD   0
024426  005367  177772                  DEC     -6(PC)
024432  001375                          BNE     .-4
024434  005367  177756                  DEC     -22(PC)
024440  001367                          BNE     .-20
024442  005237  017232                  INC     XDELAY                  ;WAIT FACTOR EXPIRED?
024446  002761                          BLT     3$                      ;BRANCH - IF NO
024450  000207                  4$:     RTS     PC                      ;RETURN

024452  012737  000160  002116  XTIME:  MOV     #160,L$DLY              ;GET OUTER DELAY LOOP
024460  006337  017234                  ASL     YDELAY                  ;MULTIPLY FACTOR BY 4
024464  006337  017234                  ASL     YDELAY                  ; ------------------
024470  005437  017234                  NEG     YDELAY                  ;GET NEGATIVE OF RESULT
024474                                  READBUS                         ;Q-BUS?
024474  104407                          TRAP    C$RDBU
024476                                  BNCOMPLETE      1$              ;BRANCH - IF NO
024476  103023                          BCC     1$
024500  012737  000150  002116          MOV     #150,L$DLY              ;GET OUTER DELAY LOOP
024506                          2$:     DELAY   20                      ;WAIT WITH RESPECT TO FONZ BUS
024506  012727  000020                  MOV     #20,(PC)+
024512  000000                          .WORD   0
024514  013727  002116                  MOV     L$DLY,(PC)+
```

GLOBAL SUBROUTINES SECTION

```
                024520  000000                  .WORD   0
                024522  005367  177772          DEC     -6(PC)
                024526  001375                  BNE     .-4
                024530  005367  177756          DEC     -22(PC)
                024534  001367                  BNE     .-20
     35         024536  005237  017234          INC     YDELAY          ;WAIT FACTOR EXPIRED?
     36         024542  002761                  BLT     2$              ;BRANCH - IF NO
     37         024544  000417                  BR      3$              ;EXIT
     38         024546                  1$:     DELAY   50              ;WAIT
                024546  012727  000050          MOV     #50,(PC)+
                024552  000000                  .WORD   0
                024554  013727  002116          MOV     L$DLY,(PC)+
                024560  000000                  .WORD   0
                024562  005367  177772          DEC     -6(PC)
                024566  001375                  BNE     .-4
                024570  005367  177756          DEC     -22(PC)
                024574  001367                  BNE     .-20
     39         024576  005237  017234          INC     YDELAY          ;WAIT FACTOR EXPIRED?
     40         024602  002761                  BLT     1$              ;BRANCH - IF NO
     41         024604  000207          3$:     RTS     PC              ;RETURN
     42
     43
```

```
                                        ;ROUTINE TO PERFORM OVERWRITE
                                        ;CALL:  JSR     R5,OVWPER
                                        ;       SECTORS TO WRITE FORWARD
                                        ;       SECTORS TO WRITE REVERSE

024606  010046                  OVWPER: MOV     R0,-(SP)        ;SAVE R0, R1, R2, R3
024610  010146                          MOV     R1,-(SP)
024612  010246                          MOV     R2,-(SP)
024614  010346                          MOV     R3,-(SP)
024616  005000                          CLR     R0              ;R0 HAS COUNT IF R0<5.
024620  012537  003126                  MOV     (R5)+,FORSK     ;USE TOP SURFACE, IF R0>5.
024624  012537  003124                  MOV     (R5)+,REVSK     ;USE BOTTOM SURFACE, IF R0>1
                                                                ;DONE.
024630  012701  003022                  MOV     #OVWTRK,R1      ;GET START OF LIST OF TRACKS
024634  011102                  1$:     MOV     (R1),R2         ;GET POINTER TO TRACK
024636  021227  177777                  CMP     (R2),#-1        ;LEGIT TRACK??????
024642  001500                          BEQ     3$              ;NO, EXIT

024644  005037  003122                  CLR     CYL             ;CLEAR CYLINDER/HEAD FOR SEEK
024650  005037  003120                  CLR     SURF
024654  020027  000005                  CMP     R0,#5           ;TOP/BOTTOM
024660  002402                          BLT     2$              ;TOP, BRANCH
024662  005237  003120                  INC     SURF            ;BOTTOM SURFACE
024666  004537  026254          2$:     JSR     R5,SKCYL        ;SEEK TO CYLINDER
024672  005037  003122                  CLR     CYL
024676  051237  003122                  BIS     (R2),CYL
024702  004537  026254                  JSR     R5,SKCYL        ;SEEK TO PROPER CYLINDER
024706  013703  003126                  MOV     FORSK,R3        ;SECTORS TO WRITE
024712  004537  025070                  JSR     R5,WRSEC        ;GO WRITE SECTORS
024716  000034                          .WORD   28.
024720  012737  020116  003116          MOV     #FWD,DIRC       ;SET FORWARD DIRECTION
024726  004537  027200                  JSR     R5,VEROW        ;VERIFY OVERWRITE
024732  004537  027564                  JSR     R5,VEROD        ;VERIFY OTHER DRIVES DATA
024736  005037  003122                  CLR     CYL
024742  022737  000001  003060          CMP     #1,T.DRIVE      ;RL01?
024750  001004                          BNE     50$             ;NO
024752  052737  000377  003122          BIS     #377,CYL        ;SET TO GO TO MAX CYL
024760  000403                          BR      51$
024762  052737  000777  003122  50$:    BIS     #777,CYL        ;MAX CYL FOR RL02
024770  004537  026254          51$:    JSR     R5,SKCYL        ;SEEK TO MAX CYLINDER ON DRIVE
024774  005037  003122                  CLR     CYL
025000  005037  003120                  CLR     SURF
025004  051237  003122                  BIS     (R2),CYL
025010  004537  026254                  JSR     R5,SKCYL        ;DO ANOTHER SEEK

025014  013703  003124                  MOV     REVSK,R3        ;SECTORS TO WRITE
025020  004537  025070                  JSR     R5,WRSEC        ;WRITE THEM
025024  000034                          .WORD   28.
025026  012737  020126  003116          MOV     #REV,DIRC       ;SET DIRECTION
025034  004537  027200                  JSR     R5,VEROW        ;VERIFY OVERWRITE
025040  004537  027564                  JSR     R5,VEROD        ;VERIFY OTHER DRIVES DATA

025044  005721                  3$:     TST     (R1)+           ;INCREMENT TO NEXT TRACK
025046  005200                          INC     R0              ;ACCOUNT FOR IT
025050  020027  000012                  CMP     R0,#10.         ;DONE?
025054  001267                          BNE     1$              ;NO, GO BACK
```

```
     58
     59 025056  012603                    MOV     (SP)+,R3        ;RESTORE REG.
     60 025060  012602                    MOV     (SP)+,R2
     61 025062  012601                    MOV     (SP)+,R1
     62 025064  012600                    MOV     (SP)+,R0
     63 025066  000205                    RTS     R5              ;EXIT
```

```
                                        ;ROUTINE TO WRITE SECTORS
                                        ;USED IN OVERWRITE TEST;ADJACENT CYLINDER TEST
                                        ;CALL   JSR     R5,WRSEC
                                        ;       .WRD                    ;STARTING SECTOR
                                        ;R3 HAS BITMAP OF SECTORS TO WRITE
                                        ;R4 HAS DRIVE BUFFER POINTER

025070  010046                 WRSEC:  MOV     R0,-(SP)        ;SAVE R0
025072  010146                         MOV     R1,-(SP)        ;SAVE R1
025074  010246                         MOV     R2,-(SP)        ;SAVE R2
025076  012701  003232                  MOV     #BUF,R1         ;WRITE PATTERN INTO
025102  012702  000200                  MOV     #128.,R2        ;MEMORY THAT WE
025106  016421  000006          2$:     MOV     PAT(R4),(R1)+   ;WILL WRITE ONTO
025112  005302                         DEC     R2              ;PACK FOR THIS
025114  001374                         BNE     2$              ;DRIVE
025116  012701  100000                  MOV     #100000,R1      ;MASK FOR BIT MAP
025122  012737  000007  003056          MOV     #7,TEM
025130  053702  003122                  BIS     CYL,R2
025134  006302                  120$:   ASL     R2
025136  005337  003056                  DEC     TEM
025142  001374                         BNE     120$
025144  005737  003120                  TST     SURF
025150  001402                         BEQ     3$              ;0, SKIP
025152  052702  000100                  BIS     #HEAD,R2        ;SET BOTTOM HEAD
025156  052502                  3$:     BIS     (R5)+,R2        ;START AT SECTOR 28.
025160  030103                  4$:     BIT     R1,R3           ;WRITE THIS SECTOR?
025162  001452                         BEQ     5$              ;NO

025164  005037  003110                  CLR     HSFLG
025170  012737  177600  003206          MOV     #-128.,BMP      ;LOAD WORD COUNT
025176  010237  003204                  MOV     R2,BDA          ;LOAD DISK ADDRESS
025202  010237  003066                  MOV     R2,TEMP         ;SAVE DISK ADDRESS
025206  042702  177700                  BIC     #177700,R2
025212  020227  000047                  CMP     R2,#39.
025216  003403                         BLE     6$
025220  162737  000050  003204          SUB     #40.,BDA
025226  012737  003232  003202  6$:     MOV     #BUF,BBA        ;LOAD BUS ADDRESS
025234  013702  003066                  MOV     TEMP,R2         ;RESTORE DISK ADDRESS
025240  004537  032404          11$:    JSR     R5,LDFUNC       ;GO WRITE
025244  000012                          WRITE
025246  005737  003074                  TST     ERFLG           ;ERROR IN WRITING
025252  001416                         BEQ     5$              ;NO,OKAY
025254  005737  003110                  TST     HSFLG
025260  001007                         BNE     10$
025262                                  ERRSOFT 100.,WRIT1,ERR2
025262  104457                         TRAP    C$ERSOFT
025264  000144                         .WORD   100
025266  020136                         .WORD   WRIT1
025270  020276                         .WORD   ERR2
025272  005237  003110                  INC     HSFLG
025276  000760                         BR      11$
025300                          10$:    ERRHRD  110.,WRIT1,ERR2
025300  104456                         TRAP    C$ERHRD
025302  000156                         .WORD   110
025304  020136                         .WORD   WRIT1
025306  020276                         .WORD   ERR2
```

GLOBAL SUBROUTINES SECTION

```
     50 025310  005202                5$:     INC     R2              ;NEXT SECTOR
     51 025312  000241                        CLC                     ;CLEAR CARRY BIT
     52 025314  006001                        ROR     R1              ;DONE?
     53 025316  103320                        BCC     4$              ;NO GO BACK
     54 025320  012602                        MOV     (SP)+,R2        ;RESTORE REGISTERS AND EXIT
     55 025322  012601                        MOV     (SP)+,R1
     56 025324  012600                        MOV     (SP)+,R0
     57 025326  000205                        RTS     R5
```

```
    1 025330  005037  003214          ADJCYL: CLR     ADJTRK          ;INSIDE/OUTSIDE TRACK FLAG
    2 025334  005037  003114                  CLR     HEAD01          ;INIT TO TOP SURFACE
    3 025340  012737  000001  003216          MOV     #1,ADJUUT       ;START OF TRACK LIST
    4 025346  012701  002236          21$:    MOV     #OUT10,R1       ;
    5 025352  012537  003100          20$:    MOV     (R5)+,ADJLOC    ;PICK UP TRACK OFFSET
    6 025356  001003                          BNE     1$              ;IS THERE ONE?
    7 025360  005037  003104                  CLR     ADJDIR
    8 025364  000205                          RTS     R5              ;NO EXIT
    9 025366  012537  003220          1$:     MOV     (R5)+,ADJLC2    ;YES, GET REST OF INFO
   10 025372  012537  003222                  MOV     (R5)+,ADJLC3
   11 025376  012537  003224                  MOV     (R5)+,ADJLC4
   12 025402  113700  003100          2$:     MOVB    ADJLOC,R0       ;GET OFFSET
   13 025406  012737  000020  003230          MOV     #16.,STSEC      ;STARTING SECTOR IS 16
   14
   15 025414  010102                          MOV     R1,R2           ;GET START INTO R2
   16
   17 025416  005300                  3$:     DEC     R0              ;DOWN COUNT OFFSET
   18 025420  001414                          BEQ     4$              ;FOUND IT?
   19
   20 025422  005722                          TST     (R2)+           ;INDEX (R2)
   21 025424  062737  000042  003230          ADD     #34.,STSEC      ;NO, NEXT SECTOR
   22 025432  022737  000050  003230          CMP     #40.,STSEC
   23 025440  003366                          BGT     3$
   24 025442  162737  000050  003230          SUB     #40.,STSEC
   25 025450  000762                          BR      3$              ;BACK FOR NEXT
   26
   27 025452  021227  177777          4$:     CMP     (R2),#-1        ;LEGAL TRACK?
   28 025456  001002                          BNE     5$              ;YES, CONTINUE
   29
   30 025460  000137  026126                  JMP     13$             ;NO PICK UP NEXT SET
   31
   32 025464  005037  003120          5$:     CLR     SURF            ;SET UP FOR OUTER TRACK
   33 025470  005037  003122                  CLR     CYL
   34
   35 025474  005737  003114                  TST     HEAD01          ;WHICH HEAD?
   36 025500  001403                          BEQ     6$              ;TOP, SKIP
   37
   38 025502  052737  000001  003120          BIS     #1,SURF ;LOWER HEAD, SET IT!
   39
   40 025510  004537  026254          6$:     JSR     R5,SKCYL        ;SEEK TO OUTER TRACK
   41
   42 025514  011237  003122                  MOV     (R2),CYL        ;GET DESIRED TRACK
   43
   44 025520  004537  026254                  JSR     R5,SKCYL        ;SEEK TO IT
   45 025524  012737  020116  003116          MOV     #FWD,DIRC       ;SEEK DIRECTION
   46 025532  113703  003101                  MOVB    ADJLOC+1,R3     ;GET SECTORS TO WRITE
   47 025536  000303                          SWAB    R3              ;ALIGN IT
   48 025540  042703  000377                  BIC     #377,R3         ;CLEAR OUT HIGH BYTE
   49
   50 025544  022737  000047  003230          CMP     #39.,STSEC      ;OVER FORTY?
   51 025552  002003                          BGE     7$              ;NO, CONTINUE
   52
   53 025554  162737  000050  003230          SUB     #40.,STSEC      ;YES BACK IT UP
   54 025562  013737  003230  025574  7$:     MOV     STSEC,8$        ;STARTING SECTOR
   55
   56 025570  004537  025070                  JSR     R5,WRSEC        ;WRITE SECTORS
   57 025574  000000                  8$:     .WORD   0
```

```
     58 025576  013737  025574  025610          MOV     8$,108$
     59 025604  004537  030112                  JSR     R5,VAJWR        ;VERIFY THIS WRITE
     60 025610  000000                  108$:   .WORD   0
     61 025612  013737  025610  025624          MOV     108$,208$
     62 025620  004537  030356                  JSR     R5,BSVWR
     63 025624  000000                  208$:   .WORD   0
     64 025626  013737  003230  003226          MOV     STSEC,STSEC1    ;GET OTHER SECTORS TC WRITE
     65 025634  062737  000010  003226          ADD     #8.,STSEC1      ;8 SECTORS GONE BY
     66 025642  022737  000047  003226          CMP     #39.,STSEC1     ;GONE PAST 40?
     67 025650  002003                          BGE     9$              ;NO, OKAY
     68
     69 025652  162737  000050  003226          SUB     #40.,STSEC1     ;YES BACK IT UP
     70
     71 025660  013703  003220          9$:     MOV     ADJLC2,R3       ;GET SECTORS TO WRITE
     72
     73 025664  013737  003226  025676          MOV     STSEC1,10$      ;STARTING SECTORS
     74
     75 025672  004537  025070                  JSR     R5,WRSEC        ;WRITE SECTORS
     76 025676  000000                  10$:    .WORD   0
     77 025700  013737  025676  025712          MOV     10$,110$
     78 025706  004537  030112                  JSR     R5,VAJWR        ;VERIFY THIS WRITE
     79 025712  000000                  110$:   .WORD   0
     80 025714  013737  025712  025726          MOV     110$,210$
     81 025722  004537  030356                  JSR     R5,BSVWR        ;VERIFY ADJ CYL + 1
     82 025726  000000                  210$:   .WORD   0
     83 025730  022737  000001  003060          CMP     #1,T.DRIVE
     84 025736  001004                          BNE     77$
     85 025740  012737  000377  003122          MOV     #377,CYL
     86 025746  000403                          BR      88$
     87
     88 025750  012737  000777  003122  77$:    MOV     #777,CYL
     89
     90 025756  004537  026254          88$:    JSR     R5,SKCYL
     91
     92 025762  011237  003122                  MOV     (R2),CYL        ;SEEK BACK TO PROPER TRACK
     93
     94 025766  004537  026254                  JSR     R5,SKCYL        ;SEEK TO PROPER CYLINDER
     95 025772  012737  020126  003116          MOV     #REV,DIRC       ;SEEK DIRECTION
     96 026000  113703  003223                  MOVB    ADJLC3+1,R3     ;GET SECTORS TO WRITE
     97
     98 026004  000303                          SWAB    R3              ;ALIGN IT
     99 026006  042703  000377                  BIC     #377,R3         ;CLEAR OUT HIGH BYTE
    100 026012  013737  003230  026024          MOV     STSEC,11$
    101
    102 026020  004537  025070                  JSR     R5,WRSEC        ;WRITE PROPER SECTOR
    103 026024  000000                  11$:    .WORD   0
    104
    105 026026  013737  026024  026040          MOV     11$,111$
    106 026034  004537  030112                  JSR     R5,VAJWR        ;VERIFY THIS WRITE
    107 026040  000000                  111$:   .WORD   0
    108 026042  013737  026040  026054          MOV     111$,211$
    109 026050  004537  030356                  JSR     R5,BSVWR
    110 026054  000000                  211$:   .WORD   0
    111 026056  013703  003224                  MOV     ADJLC4,R3       ;GET SECTORS
    112 026062  013737  003226  026074          MOV     STSEC1,12$      ;GET SECTORS TO WRITE
    113
    114 026070  004537  025070                  JSR     R5,WRSEC        ;WRITE PROPER SECTORS
```

```
                                12$:    .WORD   0

                                        MOV     12$,112$
                                        JSR     R5,VAJWR        ;VERIFY THIS WRITE
                                112$:   .WORD   0

                                        MOV     112$,212$
                                        JSR     R5,BSVWR        ;VERIFY ADJ CYLINDERS + 1
                                212$:   .WORD   0

                                13$:    TST     HEAD01          ;WHICH HEAD WERE WE DOING?
                                        BNE     14$
                                        INC     HEAD01
                                        BR      99$
                                14$:    CLR     HEAD01          ;NEXT SET OF TRACKS
                                99$:    ADD     #10.,R1         ;NEXT SET OF TRACKS
                                        CMP     R1,#INN51       ;END OF LIST
                                        BGE     18$             ;END OF TRACK LIST
                                        JMP     2$              ;NO GO BACK

                                ;AT END OF TRACK LIST NEXT GROUP OF WRITES

                                18$:    TST     FADJ            ;FIRST SET?
                                        BEQ     15$             ;NO, CONTINUE
                                        CLR     FADJ            ;YES, CLEAR FIRST
                                        BR      17$             ;EXIT
                                15$:    TST     ADJTRK          ;DONE BOTH INSIDE OUTSIDE
                                        BNE     16$             ;TRACKS, YES 16$
                                        INC     ADJTRK          ;NO, SET INSIDE FLAG
                                        JMP     21$             ;GO DO INSIDE TRACK
                                16$:    CLR     ADJTRK          ;BACK TO OUTSIDE TRACK
                                        INC     ADJUUT          ;DONE WITH ANOTHER
                                        CMP     ADJUUT,UUT      ;DONE TABLE FOR ALL UUT?
                                        BEQ     17$             ;YES, FOR EXIT
                                        JMP     21$             ;NO, GO BACK FOR NEXT
                                17$:    TST     (R5)+           ;BUMP EXIT TO END OF
                                        BNE     17$             ;TABLE FOR PROPER RETURN
                                        CLR     ADJDIR          ;EXIT
                                        RTS     R5
```

```
                                        ;ROUTINE TO SEEK TO A DESIRED CYLINDER
                                        ;CALL:  JSR     R5,SKCYL
                                        ;ROUTINE HAS DESIRED CYLINDER IN LOC "CYL"
                                        ;
                                        ;
026254  010146                  SKCYL:  MOV     R1,-(SP)        ;SAVE R1
026256  004537  032404           90$:    JSR     R5,LDFUNC       ;GET PRESENT POSITION
026262  000010                          RDHDR
                                
026264  005737  003074                  TST     ERFLG           ;ERROR FLAG SET
026270  001104                          BNE     5$              ;YES, SKIP
                                
026272  005001                          CLR     R1
026274  012737  000007  003056          MOV     #7,TEM
026302  053701  003122                  BIS     CYL,R1          ;GET THE SELECTED CYLINDER NUMBER
                                
026306  006301                  120$:   ASL     R1
026310  005337  003056                  DEC     TEM
026314  001374                          BNE     120$
026316  042737  000177  003172          BIC     #177,E.MP
026324  163701  003172                  SUB     E.MP,R1         ;CALCULATE DIFFERENCE WORD
026330  103002                          BCC     1$              ;IF POSITIVE SET DIRECTION
026332  005401                          NEG     R1              ;NEGATE
026334  000402                          BR      2$              ;SKIP SETTING DIRECTION
026336  052701  000004           1$:    BIS     #SIGN,R1        ;SET FOR FORWARD SEEK
026342  052701  000001           2$:    BIS     #MK,R1          ;SET MARKER BIT
026346  005737  003120                  TST     SURF
026352  001402                          BEQ     3$              ;TOP
026354  052701  000020                  BIS     #SKHS,R1        ;BOTTOM
026360  010137  003204           3$:    MOV     R1,BDA          ;LOAD DIFFERENCE WORD
026364  004537  032404                  JSR     R5,LDFUNC       ;EXECUTE SEEK
026370  000006                          SEEK
                                
026372  005737  003074                  TST     ERFLG           ;ERROR?
026376  001041                          BNE     5$              ;YES, SKIP
                                
026400  004537  032404                  JSR     R5,LDFUNC       ;VERIFY POSITION?
026404  000010                          RDHDR
026406  005737  003074                  TST     ERFLG
026412  001033                          BNE     5$
026414  042737  000077  003172          BIC     #77,E.MP        ;VERIFY POSITION
026422  005001                          CLR     R1
026424  012737  000007  003056          MOV     #7,TEM
026432  053701  003122                  BIS     CYL,R1
026436  006301                  220$:   ASL     R1
026440  005337  003056                  DEC     TEM
026444  001374                          BNE     220$
026446  005737  003120                  TST     SURF
026452  001402                          BEQ     4$
026454  052701  000100                  BIS     #HEAD,R1
026460  020137  003172           4$:    CMP     R1,E.MP
026464  001414                          BEQ     6$
                                
026466                                  ERRDF   12.,SKER,ERR6   ;MIS SEEK ERROR
026466  104455                          TRAP    C$ERDF
026470  000014                          .WORD   12
026472  020077                          .WORD   SKER
```

```
            026474  020704                          .WORD   ERR6
     55     026476  000137  026256                  JMP     90$
     56
     57     026502                          5$:     ERRDF   13.,FUNERR,ERR5 ;ERROR IN SEEK OPERATION
            026502  104455                          TRAP    C$ERDF
            026504  000015                          .WORD   13
            026506  020047                          .WORD   FUNERR
            026510  020644                          .WORD   ERR5
     58     026512  000137  026256                  JMP     90$
     59     026516  012601                  6$:     MOV     (SP)+,R1        ;CANT GET THERE
     60     026520  000205                          RTS     R5              ;EXIT
```

```
                                        ;ROUTINE TO PERFORM REGISTER PRINTOUT DUMP
                                        ;CALL:  JSR     PC,REGDMP
                                                                        ;PROMPT - BEFORE CS: _ BA: _ DA: _ MP: _
026522                                  REGDMP: PRINTB  #FRM12,BCS,BBA,BDA,BMP
026522  013746  003206                          MOV     BMP,-(SP)
026526  013746  003204                          MOV     BDA,-(SP)
026532  013746  003202                          MOV     BBA,-(SP)
026536  013746  003200                          MOV     BCS,-(SP)
026542  012746  021646                          MOV     #FRM12,-(SP)
026546  012746  000005                          MOV     #5,-(SP)
026552  010600                                  MOV     SP,R0
026554  104414                                  TRAP    C$PNTB
026556  062706  000014                          ADD     #14,SP
                                                                        ;PROMPT - AFTER CS: _ BA: _ DA: _ MP: _
026562                                          PRINTB  #FRM13,E.CS,E.BA,E.DA,E.MP
026562  013746  003172                          MOV     E.MP,-(SP)
026566  013746  003170                          MOV     E.DA,-(SP)
026572  013746  003166                          MOV     E.BA,-(SP)
026576  013746  003164                          MOV     E.CS,-(SP)
026602  012746  021725                          MOV     #FRM13,-(SP)
026606  012746  000005                          MOV     #5,-(SP)
026612  010600                                  MOV     SP,R0
026614  104414                                  TRAP    C$PNTB
026616  062706  000014                          ADD     #14,SP
026622  032737  040000  003164                  BIT     #BIT14,E.CS
026630  001437                                  BEQ     1$
026632  016403  000000                          MOV     CSR(R4),R3
026636  012763  000013  000004                  MOV     #13,DA(R3)
026644  012737  000004  003200                  MOV     #4,BCS
026652  056437  000004  003200                  BIS     DSB(R4),BCS
026660  013763  003200  000000                  MOV     BCS,CS(R3)
026666  032763  000200  000000          2$:     BIT     #200,CS(R3)
026674  001774                                  BEQ     2$
026676  016337  000006  003106                  MOV     MP(R3),DRSTAT
026704                                          PRINTB  #FRM14,DRSTAT           ;PROMPT - DRIVE STATUS
026704  013746  003106                          MOV     DRSTAT,-(SP)
026710  012746  022010                          MOV     #FRM14,-(SP)
026714  012746  000002                          MOV     #2,-(SP)
026720  010600                                  MOV     SP,R0
026722  104414                                  TRAP    C$PNTB
026724  062706  000006                          ADD     #6,SP
026730  000207                          1$:     RTS     PC
```

```
                                        ;ROUTINE TO STORE OR RETRIEVE ADJACENT CYLINDER SECTOR DRIVE
                                        ;INFORMATION FROM THE 24X5 "SECLST" BUFFER.
                                        ;ENTER WITH R0 = SECTOR REQUEST
                                        ;EXIT WITH R0 = ADJACENT CYLINDER DRIVE INFORMATION FOR SECTOR
                                        ;EXIT WITH R0 = 0 IF SECTOR REQUESTED IS NOT IN BUFFER MAP
                                        ;CALL 1:        JSR R5,RSADJS
                                        ;               .WORD   0               ;RETRIEVE SECTOR INFO.
                                        ;CALL 2:        JSR R5,RSADJS
                                        ;               .WORD   1               ;STORE SECTOR INFO.
026732  010146                 RSADJS: MOV     R1,-(SP)
026734  010246                         MOV     R2,-(SP)
026736  010346                         MOV     R3,-(SP)
026740  042700  177700                 BIC     #177700,R0      ;SAVE SECTOR BITS
026744  012537  003102                 MOV     (R5)+,ADJFLG    ;SAVE RETRIEVE/STORE FLAG
026750  012701  000001                 MOV     #1,R1           ;START WITH TRACK (N-2)
026754  012702  002442                 MOV     #SECBUF,R2      ;START OF 24X5 BUFFER
026760  012703  000020                 MOV     #16.,R3         ;SECTOR 16 START FOR (N-2) TRACK
026764  123701  003100         1$:     CMPB    ADJLOC,R1       ;CHECK TRACK INDEX
026770  001413                         BEQ     2$              ;
026772  005201                         INC     R1              ;INDEX TRACK REFERENCE
026774  062702  000060                 ADD     #48.,R2         ;UPDATE BUFFER TO NEXT TRACK REF.
027000  062703  000042                 ADD     #34.,R3         ;UPDATE SECTOR START FOR NEXT TRACK
027004  020327  000050                 CMP     R3,#40.
027010  002765                         BLT     1$
027012  162703  000050                 SUB     #40.,R3
027016  000762                         BR      1$              ;
027020  012701  000030         2$:     MOV     #24.,R1         ;SET COUNTER FOR 24 SECTORS
027024  020003                 3$:     CMP     R0,R3           ;COMPARE SECTOR TO SECTOR TABLE
027026  001413                         BEQ     5$              ;YES,STORE OR RETRIEVE SECTOR INFO.
027030  005722                         TST     (R2)+           ;INDEX SECLST BUFFER IN WORD FORMAT
027032  005203                         INC     R3              ;INDEX SECTOR COUNT
027034  020327  000047                 CMP     R3,#39.         ;COMPARE SECTOR COUNT FOR <40
027040  003402                         BLE     4$
027042  162703  000050                 SUB     #40.,R3         ;KEEP SECTOR COUNT<40
027046  005301                 4$:     DEC     R1              ;PASSED 24 SECTORS?
027050  001365                         BNE     3$              ;COMPARE NEXT SECTOR
027052  005000                         CLR     R0              ;SETUP R0 FOR EXIT
027054  000405                         BR      7$              ;EXIT ROUTINE,SECTOR NOT FOUND
027056  005737  003102         5$:     TST     ADJFLG          ;FLAG=0 FOR RETRIEVE
027062  001401                         BEQ     6$
027064  010412                         MOV     R4,(R2)         ;STORE DRIVE INFO. INTO BUFFER
027066  011200                 6$:     MOV     (R2),R0         ;SAVE DRIVE INFO. INTO R0 FOR EXIT
027070  012603                 7$:     MOV     (SP)+,R3
027072  012602                         MOV     (SP)+,R2
027074  012601                         MOV     (SP)+,R1
027076  000205                         RTS     R5              ;EXIT
```

```
                                        ;ROUTINE TO SET DRIVE IN SECTOR LIST
                                        ;CALL:  JSR     R5,SETLST       ;R0 HAS SECTOR
                                        ;DRIVE GOTTEN FROM R4
                                        
027100  010146                 SETLST: MOV     R1,-(SP)        ;SAVE R1
                                        
027102  162700  000034                  SUB     #28.,R0         ;START LIST AT 0
027106  100002                          BPL     3$
027110  062700  000050                  ADD     #40.,R0
027114  012701  002402          3$:     MOV     #SECLST,R1      ;BEGINNING OF SECTOR LIST
027120  005700                  1$:     TST     R0              ;FOUND SECTOR?
027122  001403                          BEQ     2$              ;BRANCH IF YES
027124  005300                          DEC     R0              ;DECREMENT SECTOR
027126  005721                          TST     (R1)+           ;NEXT ENTRY IN LIST
027130  000773                          BR      1$              ;GO BACK
027132  010411                  2$:     MOV     R4,(R1)         ;STORE DRIVE BITS IN LIST
027134  012601                          MOV     (SP)+,R1        ;RESTORE R1
027136  000205                          RTS     R5
                                        
                                        ;ROUTINE TO LOCATE DRIVE THAT WROTE SECTOR LAST
                                        ;CALL:  JSR     R5,FNDDRV       ;R0-CONTAINS SECTOR
                                        ;                               ;ON EXIT R0-DRIVE
                                        
027140  010146                 FNDDRV: MOV     R1,-(SP)        ;SAVE R1
027142  162700  000034                  SUB     #28.,R0         ;START LIST AT 0
027146  100002                          BPL     3$
027150  062700  000050                  ADD     #40.,R0
027154  012701  002402          3$:     MOV     #SECLST,R1      ;START OF LIST
027160  005700                  1$:     TST     R0              ;FOUND SECTOR?
027162  001403                          BEQ     2$              ;YES, GET DRIVE #, EXIT
027164  005300                          DEC     R0              ;NO, DOWN COUNT SECTOR
027166  005721                          TST     (R1)+           ;NEXT ENTRY IN LIST
027170  000773                          BR      1$              ;GO BACK
027172  011100                  2$:     MOV     (R1),R0         ;GET DRIVE BUFFER POINTER
027174  012601                          MOV     (SP)+,R1        ;RESTORE R1
027176  000205                          RTS     R5              ;EXIT
```

```
                                        ;
                                        ;ROUTINE TO VERIFY THAT THE OVERWRITE DID ACTUALLY OVERWRITE THE
                                        ;PREVIOUS DATA ON THE PACK.
                                        ;
                                        ;CALL:  JSR     R5,VEROW        USES R3 AS BIT MAP OF SECTORS TO
                                        ;                               CHECK.  R3 IS LOADED PRIOR TO
                                        ;                               WRITING SECTORS.

027200  010046                  VEROW:  MOV     R0,-(SP)        ;SAVE REGISTER CONTENTS
027202  010146                          MOV     R1,-(SP)
027204  010246                          MOV     R2,-(SP)
027206  012737  000034  003132          MOV     #28.,SECT               ;START VERIFY AT SECTOR 28
027214  012701  100000                  MOV     #100000,R1      ;BIT MASK FOR VERIFICATION
027220  016437  000006  003136          MOV     PAT(R4),GDATA   ;GET PATTERN FOR THIS DRIVE

027226  012737  177600  003206  1$:     MOV     #-128.,BMP      ;SET UP READ-ONE SECTOR
027234  012737  003232  003202          MOV     #BUF,BBA        ;BUS ADDRESS
027242  042737  000077  003204  2$:     BIC     #77,BDA         ;CLEAR OUT SECTOR BITS
027250  053737  003132  003204          BIS     SECT,BDA        ;SET SECTOR
027256  030103                          BIT     R1,R3           ;DO WE READ THIS ONE?
027260  001521                          BEQ     5$              ;NO, BRANCH
027262  004537  032404                  JSR     R5,LDFUNC       ;READ
027266  000014                          READ

027270  005737  003164                  TST     E.CS            ;ERROR
027274  100107                          BPL     4$              ;NO CONTINUE

027276  005737  003062                  TST     FOWR            ;INITIAL WRITE
027302  001412                          BEQ     21$             ;NO
027304  012737  017373  003072          MOV     #INITWR,REASON  ;SETUP INITIAL WRITE OF SECTOR
027312  016437  000000  003070          MOV     CSR(R4),LSTCLR
027320  016437  000005  003134          MOV     DSB+1(R4),LSTDRV
027326  000415                          BR      22$
027330  012737  017642  003072  21$:    MOV     #OVMES,REASON   ;SET MESSAGE FOR OVERWRITE
027336  013700  003132                  MOV     SECT,R0         ;FIND DRIVE THAT LAST WROTE
027342  004537  027140                  JSR     R5,FNDDRV       ;SECTOR
027346  016037  000000  003070          MOV     CSR(R0),LSTCLR  ;GET IT'S CSR
027354  116037  000005  003134          MOVB    DSB+1(R0),LSTDRV  ;GET THE DRIVE
027362                          22$:    ERRDF   13.,OVWER,ERR4  ;PRINT ERROR
027362  104455                          TRAP    C$ERDF
027364  000015                          .WORD   13
027366  020003                          .WORD   OVWER
027370  020464                          .WORD   ERR4
027372  005037  003142                  CLR     WCOUNT          ;CLEAR BAD WORD COUNT W/IN SECTOR
027376  005037  003144                  CLR     SECWRD          ;CLEAR WORD IN SECTOR
027402  012702  003232                  MOV     #BUF,R2         ;GET BUFFER START
027406  023712  003136          3$:     CMP     GDATA,(R2)      ;IS DATA CORRECT?
027412  001417                          BEQ     31$             ;YES CHECK NEXT
027414  005237  003142                  INC     WCOUNT          ;NO ACCOUNT FOR IT
027420                                  PRINTF  #FRM8,SECWRD,GDATA,(R2)
027420  011246                          MOV     (R2),-(SP)
027422  013746  003136                  MOV     GDATA,-(SP)
027426  013746  003144                  MOV     SECWRD,-(SP)
027432  012746  021437                  MOV     #FRM8,-(SP)
027436  012746  000004                  MOV     #4,-(SP)
027442  010600                          MOV     SP,R0
027444  104417                          TRAP    C$PNTF
```

```
                027446  062706  000012                  ADD     #12,SP
     47
     48 027452  005722                          31$:    TST     (R2)+           ;NEXT
     49 027454  005237  003144                          INC     SECWRD          ;NEXT
     50 027460  023727  003144  000200                  CMP     SECWRD,#128.    ;DONE WITH SECTOR?
     51 027466  001347                                  BNE     3$              ;NO GO BACK
     52
     53 027470                                          PRINTF  #FRM9,WCOUNT    ;PRINT SUMMARY
                027470  013746  003142                  MOV     WCOUNT,-(SP)
                027474  012746  021503                  MOV     #FRM9,-(SP)
                027500  012746  000002                  MOV     #2,-(SP)
                027504  010600                          MOV     SP,R0
                027506  104417                          TRAP    C$PNTF
                027510  062706  000006                  ADD     #6,SP
     54
     55 027514  013700  003132                  4$:     MOV     SECT,R0         ;SET SECTOR IN LIST TO THE
     56 027520  004537  027100                          JSR     R5,SETLST       ;CREDIT OF THIS DRIVE
     57
     58 027524  005237  003132                  5$:     INC     SECT            ;NEXT SECTOR
     59 027530  023727  003132  000050                  CMP     SECT,#40.
     60 027536  001003                                  BNE     6$
     61 027540  162737  000050  003132                  SUB     #40.,SECT
     62 027546  000241                          6$:     CLC                     ;CLEAR CARRY
     63 027550  006001                                  ROR     R1              ;NEXT BIT
     64 027552  103225                                  BCC     1$              ;IF CLEAR NEXT
     65
     66 027554  012602                                  MOV     (SP)+,R2        ;RESTORE R2-R0, EXIT
     67 027556  012601                                  MOV     (SP)+,R1
     68 027560  012600                                  MOV     (SP)+,R0
     69 027562  000205                                  RTS     R5
```

```
                                        ;ROUTINE TO VERIFY THAT A DRIVE CAN RECOVER ANOTHER DRIVE'S DATA.
                                        ;
                                        ;CALL:  JSR     R5,VEROD        USES R3 AS BIT MAP OF SECTORS TO
                                        ;                               CHECK.  R3 IS LOAD BY WRSEC (WE
                                        ;                               USE R3 COMPLIMENTED.
                                        ;
                                        ;
                                        ;
027564  010046                  VEROD:  MOV     R0,-(SP)        ;SAVE R0-R2
027566  010146                          MOV     R1,-(SP)
027570  010246                          MOV     R2,-(SP)
027572  012701  100000                  MOV     #100000,R1      ;BIT MASK FOR SECTORS
027576  012737  000034  003132          MOV     #28.,SECT               ;START WITH SECTOR 28
027604  005737  003062                  TST     FOWR            ;CHECK FOR FIRST OVERWRITE
027610  001134                          BNE     6$
                                        
027612  012737  177600  003206  1$:     MOV     #-128.,BMP      ;SET UP READ (ONE SECTOR)
027620  012737  003232  003202          MOV     #BUF,BBA        ;BUS ADDRESS
027626  042737  000077  003204  2$:     BIC     #77,BDA         ;CLEAR SECTOR BITS
027634  053737  003132  003204          BIS     SECT,BDA        ;SET IN SECTOR BITS
027642  030103                          BIT     R1,R3           ;CHECK THIS SECTOR?
027644  001103                          BNE     5$              ;NO BRANCH
                                        
027646  013700  003132                  MOV     SECT,R0         ;FIND DRIVE THAT WROTE
027652  004537  027140                  JSR     R5,FNDDRV       ;SECTOR LAST
027656  016037  000000  003070          MOV     CSR(R0),LSTCLR  ;GET CSR OF DRIVE
027664  116037  000005  003134          MOVB    DSB+1(R0),LSTDRV  ;GET DRIVE
027672  016037  000006  003136          MOV     PAT(R0),GDATA   ;GET PATTERN
                                        
027700  004537  032404                  JSR     R5,LDFUNC       ;READ
027704  000014                          READ
                                        
027706  005737  003164                  TST     E.CS            ;ERROR?
027712  100060                          BPL     5$              ;NO, NEXT SECTOR
027714  012737  017675  003072          MOV     #RECMS,REASON   ;SET READ RECOVERY MESSAGE
027722                                  ERRDF   14.,RECER,ERR4  ;REPORT ERROR
027722  104455                          TRAP    C$ERDF
027724  000016                          .WORD   14
027726  020023                          .WORD   RECER
027730  020464                          .WORD   ERR4
                                        
027732  005037  003142                  CLR     WCOUNT          ;CLEAR BAD WORD COUNT
027736  005037  003144                  CLR     SECWRD          ;CLEAR WORD W/I SECTOR
027742  012702  003232                  MOV     #BUF,R2         ;START OF BUFFER
027746  023712  003136          3$:     CMP     GDATA,(R2)      ;DATA COMPARE
027752  001417                          BEQ     4$              ;YES, CHECK NEXT
                                        
027754  005237  003142                  INC     WCOUNT          ;ACCOUNT FOR ERROR
027760                                  PRINTF  #FRMB,SECWRD,GDATA,(R2)  ;PRINT ERROR
027760  011246                          MOV     (R2),-(SP)
027762  013746  003136                  MOV     GDATA,-(SP)
027766  013746  003144                  MOV     SECWRD,-(SP)
027772  012746  021437                  MOV     #FRMB,-(SP)
027776  012746  000004                  MOV     #4,-(SP)
030002  010600                          MOV     SP,R0
030004  104417                          TRAP    C$PNTF
030006  062706  000012                  ADD     #12,SP
                                        
```

```
     030012  005722                  4$:     TST     (R2)+           ;NEXT
     030014  005237  003144                  INC     SECWRD          ;NEXT WORD IN SECTOR
     030020  023727  003144  000200          CMP     SECWRD,#128.    ;DONE?
     030026  001347                          BNE     3$              ;NO
     030030                                  PRINTF  #FRM9,WCOUNT    ;PRINT SUMMARY
     030030  013746  003142                  MOV     WCOUNT,-(SP)
     030034  012746  021503                  MOV     #FRM9,-(SP)
     030040  012746  000002                  MOV     #2,-(SP)
     030044  010600                          MOV     SP,R0
     030046  104417                          TRAP    C$PNTF
     030050  062706  000006                  ADD     #6,SP

     030054  005237  003132          5$:     INC     SECT            ;NEXT SECTOR
     030060  023727  003132  000050          CMP     SECT,#40.
     030066  00[illegible]02                 BNE     7$
     030070  00[illegible]  003132          CLR     SECT
     030074  [illegible]0241          7$:     CLC
     030076  [illegible]001                  ROR     R1              ;NEXT BIT MAP
     030100  103244                          BCC     1$

     030102  012602                  6$:     MOV     (SP)+,R2        ;RESTORE R2-R0, EXIT
     030104  012601                          MOV     (SP)+,R1
     030106  012600                          MOV     (SP)+,R0
     030110  000205                          RTS     R5
```

```
                                        ;ROUTINE TO VERIFY THE ADJ. CYL. WRITE IS GOOD
                                        ;USES R3 AND WORD FOLLOWING CALL
                                        ;IF WRITE WAS GOOD,SECTOR WILL BE STORED IN MAP
                                        ;USING RSADJS/.WORD 1

030112  010046                  VAJWR:  MOV     R0,-(SP)        ;SAVE REGISTERS
030114  010146                          MOV     R1,-(SP)
030116  010246                          MOV     R2,-(SP)
030120  012701  100000                  MOV     #100000,R1      ;BIT MASK FOR CYLINDER
030124  012502                          MOV     (R5)+,R2        ;STARTING SECTOR
030126  005000                          CLR     R0
030130  053700  003122                  BIS     CYL,R0
030134  012737  0000C7  003056          MOV     #7,TEM

030142  006300                  2$:     ASL     R0
030144  005337  003056                  DEC     TEM
030150  001374                          BNE     2$
030152  005737  003120                  TST     SURF
030156  001402                          BEQ     3$
030160  052700  000100                  BIS     #HEAD,R0
030164  050200                  3$:     BIS     R2,R0
030166  030103                  4$:     BIT     R1,R3
030170  001462                          BEQ     5$
030172  012737  177600  003206          MOV     #-128.,BMP
030200  010037  003204                  MOV     R0,BDA
030204  010037  003066                  MOV     R0,TEMP
030210  042700  177700                  BIC     #177700,R0
030214  020027  000047                  CMP     R0,#39.
030220  003406                          BLE     6$
030222  162737  000050  003204          SUB     #40.,BDA
030230  162737  000050  003066          SUB     #40.,TEMP
030236  012737  003232  003202  6$:     MOV     #BUF,BBA
030244  005037  003110                  CLR     HSFLG
030250  013700  003066                  MOV     TEMP,R0
030254  004537  032404          10$:    JSR     R5,LDFUNC       ;READ FUNCTION
030260  000014                          READ
030262  005737  003074                  TST     ERFLG
030266  001416                          BEQ     7$
030270  005737  003110                  TST     HSFLG
030274  001007                          BNE     11$
030276                                  ERRSOFT 120.,READ1,ERR2
030276  104457                          TRAP    C$ERSOFT
030300  000170                          .WORD   120
030302  020163                          .WORD   READ1
030304  020276                          .WORD   ERR2
030306  005237  003110                  INC     HSFLG
030312  000760                          BR      10$
030314                          11$:    ERRHRD  130.,READ1,ERR2
030314  104456                          TRAP    C$ERHRD
030316  000202                          .WORD   130
030320  020163                          .WORD   READ1
030322  020276                          .WORD   ERR2
030324  010046                  7$:     MOV     R0,-(SP)
030326  004537  026732                  JSR     R5,RSADJS       ;STORE ADJ. CYL. SECTOR INFO.
030332  000001                          .WORD   1
030334  012600                          MOV     (SP)+,R0        ;RESTORE R0
030336  005200                  5$:     INC     R0
```

```
     50 030340  000241                    CLC
     51 030342  006001                    ROR     R1
     52 030344  103310                    BCC     4$
     53 030346  012602                    MOV     (SP)+,R2        ;RESTORE REGISTERS AND EXIT
     54 030350  012601                    MOV     (SP)+,R1
     55 030352  012600                    MOV     (SP)+,R0
     56 030354  000205                    RTS     R5
     57
```

```
                                        ;ROUTINE TO VERIFY THAT WRITE DID NOT DISTURB ADJACENT TRACKS
                                        ;WRITTEN BY OTHER DRIVES.
                                        ;CALL   JSR R5,BSVWR
                                        ;       .WORD                   ;STARTING SECTOR
                                        ;
                                        ;USES "ADJLOC" TO GET +1/-1 CYLINDER OFFSET
                                        ;USES R3 FOR SECTOR MAP, USES MAP AT "SECBUF" FOR INFO

030356  010046                  BSVWR:  MOV     R0,-(SP)        ;SAVE REGISTERS
030360  010146                          MOV     R1,-(SP)
030362  010246                          MOV     R2,-(SP)
030364  013746  003122                  MOV     CYL,-(SP)
030370  013746  003120                  MOV     SURF,-(SP)
030374  012546                          MOV     (R5)+,-(SP)     ;GET STARTING SECTOR
030376  123727  003100  000003          CMPB    ADJLOC,#3       ;ON MIDDLE TRACK???
030404  001455                          BEQ     BSEXIT          ;YES, THEN NO CHECK
030406  162716  000042                  SUB     #34.,(SP)               ;SETUP SECTOR START FOR OUTSIDE
030412  100002                          BPL     1$              ;IF POSITIVE OKAY ELSE FIX
030414  062716  000050                  ADD     #40.,(SP)       ;FIX IT
030420  123727  003100  000001  1$:     CMPB    ADJLOC,#1       ;ON OUTER LIMIT???
030426  001412                          BEQ     INAWR           ;YES,SKIP CHECK
030430  105337  003100                  DECB    ADJLOC          ;OUTER ADJ TRACK
030434  005337  003122                  DEC     CYL
030440  004537  030566                  JSR     R5,CHECK        ;GO CHECK ADJ SECTORS
030444  005237  003122                  INC     CYL             ;FIX BACK
030450  105237  003100                  INCB    ADJLOC
030454  062716  000104          INAWR:  ADD     #68.,(SP)       ;INNER SECTOR START
030460  021627  000050                  CMP     (SP),#40.       ;WITHIN LIMITS???
030464  002407                          BLT     1$              ;YES, OKAY
030466  162716  000050                  SUB     #40.,(SP)       ;FIX SECTOR
030472  021627  000050                  CMP     (SP),#40.
030476  002402                          BLT     1$
030500  162716  000050                  SUB     #40.,(SP)
030504  123727  003100  000005  1$:     CMPB    ADJLOC,#5       ;INNER LIMIT??
030512  001412                          BEQ     BSEXIT          ;YES,SKIP CHECK
030514  105237  003100                  INCB    ADJLOC          ;FIX FOR INNER
030520  005237  003122                  INC     CYL             ;
030524  004537  030566                  JSR     R5,CHECK        ;GO CHECK ADJ SECTORS
030530  105337  003100                  DECB    ADJLOC          ;FIX BACK
030534  005337  003122                  DEC     CYL             ;
030540  005726                  BSEXIT: TST     (SP)+           ;THROW OFF SECTOR
030542  012637  003120                  MOV     (SP)+,SURF
030546  012637  003122                  MOV     (SP)+,CYL
030552  012602                  NCHECK: MOV     (SP)+,R2
030554  012601                          MOV     (SP)+,R1
030556  012600                          MOV     (SP)+,R0
030560  004537  026254                  JSR     R5,SKCYL        ;SEEK BACK
030564  000205                          RTS     R5              ;RETURN
```

```
                                        ;ROUTINE TO VERIFY AN ADJACENT SECTOR
                                        ;CALLED FROM BSVWR
                                        ;CALL   JSR     R5,CHECK
                                        ;
                                        ;
030566  012701  100000          CHECK:  MOV     #100000,R1      ;SECTOR MASK
030572  004537  026254                  JSR     R5,SKCYL        ;GET TO DESIRED CYLINDER
030576  005002                          CLR     R2              ;CREATE ADDRESS
030600  053702  003122                  BIS     CYL,R2
030604  012737  000007  003056          MOV     #7,TEM
030612  006302                  2$:     ASL     R2
030614  005337  003056                  DEC     TEM
030620  001374                          BNE     2$
030622  005737  003120                  TST     SURF
030626  001402                          BEQ     3$              ;NO
030630  052702  000100                  BIS     #HEAD,R2
030634  056602  000002          3$:     BIS     2(SP),R2        ;SET IN SECTOR
030640  030103                  4$:     BIT     R1,R3           ;THIS SECTOR IN LIST???
030642  001452                          BEQ     5$              ;NO, NEXT
030644  010200                          MOV     R2,R0           ;COPY SECTOR
030646  042700  177700                  BIC     #177700,R0      ;ONLY SECTOR LEFT
030652  020027  000050                  CMP     R0,#40. ;SECTOR OKAY???
030656  002404                          BLT     6$              ;YES
030660  162700  000050                  SUB     #40.,R0
030664  162702  000050                  SUB     #40.,R2         ;FIX SECTOR
030670  004537  026732          6$:     JSR     R5,RSADJS       ;FIND IF SECTOR PREVIOUSLY WRITTEN
030674  000000                          .WORD   0
030676  005700                          TST     R0              ;WAS IT??
030700  001433                          BEQ     5$              ;NO
030702  010237  003204                  MOV     R2,BDA          ;LOAD DISK ADDRESS
030706  012737  177600  003206          MOV     #-128.,BMP      ;LOAD WC
030714  004537  032404                  JSR     R5,LDFUNC       ;LOAD
030720  000014                          READ
030722  005737  003074                  TST     ERFLG           ;WAS READ GOOD
030726  001420                          BEQ     5$
030730  010346                          MOV     R3,-(SP)
030732  010237  003132                  MOV     R2,SECT
030736  010003                          MOV     R0,R3
030740  042737  177700  003132          BIC     #177700,SECT
030746                                  ERRHRD  140.,ADJTXT,ERR3
030746  104456                          TRAP    C$ERHRD
030750  000214                          .WORD   140
030752  020210                          .WORD   ADJTXT
030754  020336                          .WORD   ERR3
030756  012603                          MOV     (SP)+,R3
030760                                  ERRHRD  110.,READ1,ERR2
030760  104456                          TRAP    C$ERHRD
030762  000156                          .WORD   110
030764  020163                          .WORD   READ1
030766  020276                          .WORD   ERR2
030770  005202                  5$:     INC     R2              ;NEXT SECTOR
030772  000241                          CLC
030774  006001                          ROR     R1              ;SHIFT MASK
030776  103320                          BCC     4$
031000  000205                          RTS     R5
```

```
                                        ;ROUTINE TO MERGE BAD SECTOR FILES
                                        ;ENTRY INTO THIS ROUTINE WILL OCCUR AFTER THE "SERNUM" ROUTINE
                                        ;IS PERFORMED. THE FACTORY BAD SECTOR FILE WILL BE LOCATED IN
                                        ;FIRST 400(8) LOCATIONS.
                                        ;THIS ROUTINE WILL STORE THE FIELD BAD SECTORS INTO THE NEXT
                                        ;400 LOCATIONS AND THEN MERGE THE FACTORY BAD FILE
                                        ;WITH THE FIELD BAD FILE.

                                        ;FACTORY BAD AT BUF
                                        ;FIELD BAD AT BUF + 512.

031002  010146                  MERGE:  MOV     R1,-(SP)        ;SAVE R1, R2, R3
031004  010246                          MOV     R2,-(SP)
031006  010346                          MOV     R3,-(SP)
031010  012737  003632  003202          MOV     #BUF+400,BBA    ;BUFFER START FOR FIELD BAD
031016  022737  000001  003060          CMP     #1,T.DRIVE
031024  001004                          BNE     55$
031026  012737  077724  003204          MOV     #77724,BDA
031034  000403                          BR      66$
031036  012737  177724  003204  55$:    MOV     #177724,BDA

031044  012737  177400  003206  66$:    MOV     #-256.,BMP
031052  004537  032404          97$     JSR     R5,LDFUNC       ;LOAD READ FUNCTION
031056  000014                          READ
031060  005737  003074                  TST     ERFLG           ;TEST ERROR FLAG
031064  001431                          BEQ     98$             ;YES:MERGE BAD SECTOR FILES
031066  062737  000004  003204          ADD     #4,BDA          ;TRY NEXT FIELD BAD SECTOR FILE
031074  022737  000001  003060          CMP     #1,T.DRIVE
031102  001004                          BNE     400$
031104  022737  077750  003204          CMP     #77750,BDA
031112  001357                          BNE     97$

031114  022737  177750  003204  400$:   CMP     #177750,BDA
031122  001353                          BNE     97$             ;NO,DO NEXT FIELD BAD SECTOR
031124                                  PRINTF  #FRM15
031124  012746  022037                  MOV     #FRM15,-(SP)
031130  012746  000001                  MOV     #1,-(SP)
031134  010600                          MOV     SP,R0
031136  104417                          TRAP    C$PNTF
031140  062706  000004                  ADD     #4,SP
031144                          999$:   BREAK
031144  104422                          TRAP    C$BRK
031146  000776                          BR      999$
031150  012701  003242          98$:    MOV     #BUF+10,R1      ;GET PAST ID ETC.
031154  012702  000176                  MOV     #126.,R2        ;MAX = 126
031160  005721                  1$:     TST     (R1)+           ;SECTOR OR END
031162  100404                          BMI     2$              ;END, GO GET FIELD
031164  005721                          TST     (R1)+           ;REST OF SECTOR
031166  005302                          DEC     R2              ;MAX REACHED
031170  001373                          BNE     1$              ;NO, KEEP GOING
031172  000401                          BR      3$              ;YES, SKIP BACK UP
031174  005741                  2$:     TST     -(R1)           ;BACK UP PAST TERMINATOR
031176  012703  000176          3$:     MOV     #126.,R3        ;SET 126 MAX
031202  012702  003642                  MOV     #BUF+410,R2     ;GET FIELD SECTORS
031206  012221                  4$:     MOV     (R2)+,(R1)+     ;MERGE AT END OF FACTORY
031210  100403                          BMI     5$              ;DONE?
031212  012221                          MOV     (R2)+,(R1)+     ;NO, MERGE REST OF SECTOR
```

GLOBAL SUBROUTINES SECTION

```
     52 031214  005303                    DEC     R3              ;DONE
     53 031216  001373                    BNE     4$              ;NO, GO BACK
     54 031220  012603              5$:   MOV     (SP)+,R3        ;RESTORE R3, R2, R1
     55 031222  012602                    MOV     (SP)+,R2
     56 031224  012601                    MOV     (SP)+,R1
     57 031226  000205                    RTS     R5              ;EXIT
```

```
031230  012537  003146          FNDTRK: MOV     (R5)+,OFFSET    ;GET INCREMENT/DECREMENT
031234  012537  003156                  MOV     (R5)+,SURFACE   ;GET HEAD (SURFACE)
031240  022737  000001  003060          CMP     #1,T.DRIVE
031246  001001                          BNE     80$
031250  000401                          BR      90$
031252  022525                  80$:    CMP     (R5)+,(R5)+
031254  012537  003152          90$:    MOV     (R5)+,FRTTRK
031260  012537  003150                  MOV     (R5)+,LSTTRK
031264  005037  003160                  CLR     TRKFND          ;CLEAR OUT FLAG FOUND
031270  005037  003162                  CLR     TRKCNT          ;CLEAR OUT TRACK COUNT
031274  013737  003152  003154          MOV     FRTTRK,PRSTRK   ;GET FIRST TRACK
031302                          1$:
031302  004537  031402                  JSR     R5,FNDBSC       ;IS TRACK IN BAD SECTOR FILE
031306  005737  002234                  TST     HDRFND          ;WAS IT?
031312  001003                          BNE     2$              ;YES, CLEAR TRKCNT
031314  005237  003162                  INC     TRKCNT          ;NO, INDICATE GOOD TRACK
031320  000402                          BR      3$              ;CONTINUE
031322  005037  003162          2$:     CLR     TRKCNT          ;START COUNT OVER
031326  023727  003162  000005  3$:     CMP     TRKCNT,#5       ;FIND 5 TRACKS YET?
031334  001011                          BNE     4$              ;NO, CONTINUE
031336  005237  003160                  INC     TRKFND          ;YES, EXIT WITH GOOD FLAG
031342  022737  000001  003060          CMP     #1,T.DRIVE
031350  001002                          BNE     81$
031352  062705  000004                  ADD     #4,R5

031356  000205                  81$:    RTS     R5
031360  023737  003154  003150  4$:     CMP     PRSTRK,LSTTRK   ;ARE WE DONE?
031366  001001                          BNE     5$              ;NO, KEEP LOOKING
031370  000205                          RTS     R5              ;EXIT WITH NOT FOUND
031372  063737  003146  003154  5$:     ADD     OFFSET,PRSTRK   ;NEXT TRACK
031400  000740                          BR      1$
```

```
                                        ;ROUTINE TO FIND BAD TRACK IN FILE
                                        ;CALL   JSR     R5,FNDBSC
                                        ;
031402  005037  002234        FNDBSC: CLR     HDRFND          ;INITIALIZE FLAG
031406  010146                        MOV     R1,-(SP)        ;SAVE R1, R2
031410  010246                        MOV     R2,-(SP)
031412  012701  003242                MOV     #BUF+10,R1      ;SETUP FOR BEGINNING OF FILE
031416  005711                1$:     TST     (R1)            ;END?
031420  100421                        BMI     2$              ;IF MINUS AT END, EXIT
031422  023721  003154                CMP     PRSTRK,(R1)+    ;CYLINDER CORRECT?
031426  001011                        BNE     3$              ;NO, NEXT
031430  105724                        TSTB    (R4)+           ;UPPER HALF OF WORD
031432  123711  003156                CMPB    SURFACE,(R1)    ;CORRECT SURFACT
031436  001402                        BEQ     4$
031440  105744                        TSTB    -(R4)
031442  000403                        BR      3$
031444  005237  002234        4$:     INC     HDRFND          ;SET FOUND
031450  000405                        BR      2$
                                        ;
031452  005721                3$:     TST     (R1)+           ;NEXT WORD
031454  005202                        INC     R2              ;ACCOUNT FOR IT
031456  020227  000374                CMP     R2,#252.        ;DONE?
031462  001355                        BNE     1$              ;NO, KEEP CHECKING
031464  012601                2$:     MOV     (SP)+,R1        ;RESTORE R2, R1, EXIT
031466  012602                        MOV     (SP)+,R2
031470  000205                        RTS     R5
                                        ;
031472  013701  003154        FIXCYL: MOV     PRSTRK,R1       ;GET TRACK WHICH IS GOOD
031476  005737  003146                TST     OFFSET          ;WHICH WAY WERE WE LOOKING
031502  100402                        BMI     1$              ;IN WORD, BRANCH
031504  162701  000004                SUB     #4,R1           ;BACK IT UP BY FOUR
031510  012702  000005        1$:     MOV     #5,R2           ;GOING STORE AWAY 5 TRACKS
031514  010120                2$:     MOV     R1,(R0)+        ;STORE THEM 1 WD/PER
031516  005201                        INC     R1
031520  005302                        DEC     R2
031522  001374                        BNE     2$
031524  000205                        RTS     R5
```

GLOBAL SUBROUTINES SECTION

```
                                        ;ROUTINE TO GET SERIAL NUMBER
                                        ;
                                        ;CALL   JSR     R5,SERNUM
                                        ;
031526  012737  000013  003204  SERNUM: MOV     #13,BDA
031534  004537  032404                  JSR     R5,LDFUNC       ;GET STATUS
031540  000004                          GSTAT
031542  004537  032404                  JSR     R5,LDFUNC       ;READ HEADER
031546  000010                          RDHDR
031550  013700  003172                  MOV     E.MP,R0         ;GET THE HEADER
031554  042700  000077          1$:     BIC     #77,R0          ;CLEAR SECTOR BITS
031560  022737  000001  003060          CMP     #1,T.DRIVE
031566  001003                          BNE     23$
031570  020027  077700                  CMP     R0,#77700
031574  001446                          BEQ     2$
031576  020027  177700          23$:    CMP     R0,#177700
031602  001443                          BEQ     2$
031604  042700  000100                  BIC     #100,R0         ;CLEAR HEAD
031610  022737  000001  003060          CMP     #1,T.DRIVE
031616  001003                          BNE     32$
031620  012701  077600                  MOV     #77600,R1
031624  000402                          BR      33$
031626  012701  177600          32$:    MOV     #177600,R1
                                        ;
031632  160001                  33$:    SUB     R0,R1
031634  010137  003204                  MOV     R1,BDA          ;SET UP DIF WORD
031640  052737  000025  003204          BIS     #25,BDA         ;SEEK IN, HEAD 1
031646  004537  032404                  JSR     R5,LDFUNC       ;SEEK
031652  000006                          SEEK
031654  004537  032404                  JSR     R5,LDFUNC       ;VERIFY POSITION
031660  000010                          RDHDR
031662  013700  003172                  MOV     E.MP,R0         ;GET HEADER
031666  022737  000001  003060          CMP     #1,T.DRIVE
031674  001003                          BNE     42$
031676  022700  077700                  CMP     #77700,R0
031702  000402                          BR      43$
031704  022700  177700          42$:    CMP     #177700,R0
                                        ;
031710  103321                  43$:    BHIS    1$
031712  022737  000001  003060  2$:     CMP     #1,T.DRIVE
031720  001004                          BNE     52$
031722  012737  077700  003204          MOV     #77700,BDA
031730  000403                          BR      97$
                                        ;
031732  012737  177700  003204  52$:    MOV     #177700,BDA
031740  012737  003232  003202  97$:    MOV     #BUF,BBA
031746  012737  177400  003206          MOV     #-256.,BMP
031754  004537  032404                  JSR     R5,LDFUNC       ;READ
031760  000014                          READ
031762  005737  003074                  TST     ERFLG           ;TEST ERROR FLAG
031766  001421                          BEQ     98$             ;YES,COMPARE SERIAL NUMBERS
031770  062737  000004  003204          ADD     #4,BDA          ;NO,SETUP FOR NEXT FACTORY BAD SECTOR
031776  022737  000001  003060          CMP     #1,T.DRIVE
032004  001005                          BNE     2$
032006  022737  077724  003204          CMP     #77724,BDA
032014  001351                          BNE     97$
032016  000453                          BR      99$
```

```
     58 032020  022737  177724  003204  62$:    CMP     #177724,BDA
     59 032026  001344                          BNE     97$             ;GET NEXT FACTORY BAD SECTOR
     60 032030  000446                          BR      99$             ;REPORT ERROR
     61 032032  012701  003232          98$:    MOV     #BUF,R1         ;COMPARE SERIAL NUMBERS
     62 032036  005737  003210                  TST     SERNM1          ;HAVE WE GOT ONE TO COMPARE
     63 032042  100005                          BPL     3$              ;YES, BRANCH
     64 032044  011137  003210                  MOV     (R1),SERNM1     ;NO, CALL THIS ONE IT
     65 032050  016137  000002  003212          MOV     2(R1),SERNM2    ;
     66 032056  021137  003210          3$:     CMP     (R1),SERNM1     ;SERNUM OKAY
     67 032062  001004                          BNE     4$              ;NO, PRINT ERROR
     68 032064  026137  000002  003212          CMP     2(R1),SERNM2    ;OTHER HALF OKAY
     69 032072  001437                          BEQ     5$              ;YES, EXIT
     70 032074                          4$:     PRINTF  #FRM3,2(R1),(R1),SERNM2,SERNM1
        032074  013746  003210                  MOV     SERNM1,-(SP)
        032100  013746  003212                  MOV     SERNM2,-(SP)
        032104  011146                          MOV     (R1),-(SP)
        032106  016146  000002                  MOV     2(R1),-(SP)
        032112  012746  021167                  MOV     #FRM3,-(SP)
        032116  012746  000005                  MOV     #5,-(SP)
        032122  010600                          MOV     SP,R0
        032124  104417                          TRAP    C$PNTF
        032126  062706  000014                  ADD     #14,SP
     71 032132  004537  032174                  JSR     R5,UNLOAD       ;LET OPERATOR CHANGE
     72 032136  004537  032300                  JSR     R5,LOAD         ;PACK
     73 032142  000137  031526                  JMP     SERNUM          ;GO CHECK IT AGAIN.
     74 032146                          99$:    PRINTF  #FRM15          ;MESSAGE
        032146  012746  022037                  MOV     #FRM15,-(SP)
        032152  012746  000001                  MOV     #1,-(SP)
        032156  010600                          MOV     SP,R0
        032160  104417                          TRAP    C$PNTF
        032162  062706  000004                  ADD     #4,SP
     75 032166                          999$:   BREAK
        032166  104422                          TRAP    C$BRK
     76 032170  000776                          BR      999$
     77 032172  000205                  5$:     RTS     R5
```

```
                                        ;ROUTINE UNLOAD
                                        ;CALL   JSR     R5,UNLOAD
                                                                        ;PROMPT - UNLOAD DRIVE _ ON CONTROLLER _
032174                                  UNLOAD: PRINTF  #FRM1,<B,DSB+1(R4)>,CSR(R4)     ;AND REMOVE PACK
032174  016446  000000                        MOV     CSR(R4),-(SP)
032200  005046                                CLR     -(SP)
032202  156416  000005                        BISB    DSB+1(R4),(SP)
032206  012746  020772                        MOV     #FRM1,-(SP)
032212  012746  000003                        MOV     #3,-(SP)
032216  010600                                MOV     SP,R0
032220  104417                                TRAP    C$PNTF
032222  062706  000010                        ADD     #10,SP
032226  012701  000074                        MOV     #60.,R1         ;SETUP 60 SECOND TIMER
032232  012700  000200                        MOV     #200,R0
032236  056400  000004                        BIS     DSB(R4),R0
032242  010074  000000                        MOV     R0,@CSR(R4)
032246  032774  000001  000000        2$:     BIT     #DRDY,@CSR(R4)  ;CHECK DRDY FOR ZERO
032254  001410                                BEQ     3$              ;PACK UNLOADED
032256                                        WAITMS  #10.            ;WAIT 1 SECOND
032270  005301                                DEC     R1              ;HAS 60 SEC PASSED?
032272  001365                                BNE     2$              ;NO, RETEST DRDY, CONTINUE WAIT
032274  000737                                BR      UNLOAD          ;YES, REPEAT MESSAGE CONTINUE WAIT
032276  000205                        3$:     RTS     R5              ;RETURN WITH PACK UNLOADED

                                        ;ROUTINE LOAD
                                        ;CALL   JSR     R5,LOAD
                                                                        ;PLACE PACK IN DRIVE _ ON CONTROLLER _ AND
032300                                  LOAD:   PRINTF  #FRM2,<B,DSB+1(R4)>,CSR(R4)     ;LOAD IT
032300  016446  000000                        MOV     CSR(R4),-(SP)
032304  005046                                CLR     -(SP)
032306  156416  000005                        BISB    DSB+1(R4),(SP)
032312  012746  021067                        MOV     #FRM2,-(SP)
032316  012746  000003                        MOV     #3,-(SP)
032322  010600                                MOV     SP,R0
032324  104417                                TRAP    C$PNTF
032326  062706  000010                        ADD     #10,SP
032332  012701  000170                        MOV     #120.,R1                ;SETUP 120 SEC TIMER
032336  012700  000200                        MOV     #200,R0         ;SETUP CONTROLLER READY BIT
032342  056400  000004                        BIS     DSB(R4),R0      ;SELECT DRIVE
032346  010074  000000                        MOV     R0,@CSR(R4)
032352  032774  000001  000000        2$:     BIT     #DRDY,@CSR(R4)
032360  001010                                BNE     3$
032362                                        WAITMS  #10.
032374  005301                                DEC     R1
032376  001365                                BNE     2$
032400  000737                                BR      LOAD

032402  000205                        3$:     RTS     R5
```

```
                                        ;ROUTINE LDFUNC
                                        ;CALL   JSR     R5,LDFUNC

032404  010046                  LDFUNC: MOV     R0,-(SP)
032406  010346                          MOV     R3,-(SP)
032410  010146                          MOV     R1,-(SP)
032412  005037  003074                  CLR     ERFLG           ;CLEAR ERROR FLAG
032416  016403  000000                  MOV     CSR(R4),R3      ;GET CSR
032422  013763  003206  000006          MOV     BMP,MP(R3)      ;LOAD MULTIPURPOSE
032430  013763  003204  000004          MOV     BDA,DA(R3)      ;LOAD DISK ADDRESS
032436  013763  003202  000002          MOV     BBA,BA(R3)      ;LOAD BUS ADDRESS
032444  011537  003200                  MOV     (R5),BCS        ;GET FUNCTION TO LOAD
032450  056437  000004  003200          BIS     DSB(R4),BCS     ;SELECT BITS
032456  012701  000031                  MOV     #25.,R1         ;SET WATCHDOG TO 250MS
032462  052737  000200  003200          BIS     #200,BCS
032470  013763  003200  000000          MOV     BCS,CS(R3)      ;LOAD FUNCTION
032476  016337  000000  003200          MOV     CS(R3),BCS
032504  042763  000200  000000          BIC     #200,CS(R3)
032512  032763  000200  000000  1$:     BIT     #200,CS(R3)     ;CNTLR READY?
032520  001036                          BNE     2$              ;YES, GO
032522                                  WAITUS  #100.           ;WAIT 10 MILLISECONDS
032534  005301                          DEC     R1
032536  001365                          BNE     1$

032540  016337  000000  003164          MOV     CS(R3),E.CS     ;READ ALL REGISTERS
032546  016337  000002  003166          MOV     BA(R3),E.BA
032554  016337  000004  003170          MOV     DA(R3),E.DA
032562  016337  000006  003172          MOV     MP(R3),E.MP
032570  016337  000006  003174          MOV     MP(R3),E.MP1
032576  016337  000006  003176          MOV     MP(R3),E.MP2
032604                                  ERRDF   210.,CNTTOT,ERR5;CNTLR TIMEOUT
032604  104455                          TRAP    C$ERDF
032606  000322                          .WORD   210
032610  017346                          .WORD   CNTTOT
032612  020644                          .WORD   ERR5
032614  000425                          BR      4$

032616  016337  000000  003164  2$:     MOV     CS(R3),E.CS     ;READ ALL REGISTERS
032624  016337  000002  003166          MOV     BA(R3),E.BA
032632  016337  000004  003170          MOV     DA(R3),E.DA
032640  016337  000006  003172          MOV     MP(R3),E.MP
032646  016337  000006  003174          MOV     MP(R3),E.MP1
032654  016337  000006  003176          MOV     MP(R3),E.MP2

032662  005737  003164                  TST     E.CS            ;ANY ERRORS?
032666  100002                          BPL     3$              ;YES, GO SERVICE
032670  005237  003074          4$:     INC     ERFLG
032674  005725                  3$:     TST     (R5)+
032676  012601                          MOV     (SP)+,R1
032700  012603                          MOV     (SP)+,R3
032702  012600                          MOV     (SP)+,R0
032704  000205                          RTS     R5

032706                                  ENDMOD
                                        .SBTTL CONTROL ROUTINE
```

```
                                        BGNMOD  HRDWTST
                                BGNTST

                                ;CONTROL SECTION COMPATIBILITY PROGRAM
                                ;PRINT UNLOAD AND LOAD DRIVE MESSAGES
                                ;PERFORM SERIAL CHECK ROUTINE
                                ;PERFORM READ/WRITE CHECKS ON DRIVES

032706  012701  002442          COMPAT: MOV     #SECBUF,R1      ;ADJ. CYLINDER BUFFER
032712  012700  000170                  MOV     #120.,R0        ;ADJ. CYLINDER BUFFER COUNT
032716  005021                  4$:     CLR     (R1)+           ;CLEAR ADJ. CYL. BUFFER AT STARTUP
032720  005300                          DEC     R0              ;BUFFER CLEARED?
032722  001375                          BNE     4$              ;CLEAR NEXT BUFFER WORD
032724  005237  003062                  INC     FOWR            ;SET FIRST OVERWRITE FLAG
032730  004537  024606                  JSR     R5,OVWPER       ;PERFORM OVERWRITE ON FIRST DRIVE
032734  177400                          177400
032736  000377                          377
032740  005037  003062                  CLR     FOWR            ;CLEAR FIRST OVERWRITE
032744  005237  003064                  INC     FADJ            ;SET FIRST ADJ. FLAG
032750  005237  003104                  INC     ADJDIR          ;UP = 1
032754  004537  025330                  JSR     R5,ADJCYL
032760     003     377                  .BYTE   3,377           ;TRACK AND SECTORS FOR
032762  170000                          .WORD   170000          ;INWARD SEEK
032764     003     000                  .BYTE   3,0             ;TRACK AND SECTORS FOR
032766  007777                          .WORD   7777            ;OUTWARD SEEK
032770  000000                          .WORD   0.              ;TERMINATOR
032772  004537  032174                  JSR     R5,UNLOAD       ;UNLOAD PACK FROM DRIVE UNIT
032776  062704  000010                  ADD     #PAT+2,R4       ;UPDATE POINTER FOR NEXT DRIVE
033002  004537  032300                  JSR     R5,LOAD         ;LOAD INTO SECOND DRIVE UNIT
033006  004537  031526                  JSR     R5,SERNUM       ;CHECK PACK SERIAL NUMBER
033012  004537  024606                  JSR     R5,OVWPER       ;PERFORM R/W OVERWRITE
033016  000360                          360
033020  000017                          17
033022  005237  003104                  INC     ADJDIR
033026  004537  025330                  JSR     R5,ADJCYL
033032     002     360                  .BYTE   2,360           ;IN I/O OUTSIDE
033034  000000                          .WORD   0
033036     002     017                  .BYTE   2,17            ;OUT I/O OUTSIDE
033040  000000                          .WORD   0
033042     004     360                  .BYTE   4,360           ;IN I/O INSIDE
033044  000000                          .WORD   0
033046     004     017                  .BYTE   4,17            ;OUT I/O INSIDE
033050  000000                          .WORD   0
033052  000000                          .WORD   0
033054  004537  032174                  JSR     R5,UNLOAD       ;UNLOAD PACK FROM DRIVE UNIT
033060  023727  003130  000002          CMP     UUT,#2          ;CHECK FOR > 2 DRIVES
033066  001002                          BNE     10$             ;YES,GO TO NEXT DRIVE
033070  000137  033504                  JMP     2$              ;GO TO FIRST DRIVE
033074  062704  000010          10$:    ADD     #PAT+2,R4       ;UPDATE DRIVE BUFFER FOR THIRD DRIVE
033100  004537  032300                  JSR     R5,LOAD         ;LOAD PACK FOR THIRD DRIVE
033104  004537  031526                  JSR     R5,SERNUM       ;CHECK SERIAL NUMBERS
033110  004537  024606                  JSR     R5,OVWPER       ;PERFORM R/W OVERWRITE ON THIRD DRIVE
033114  006014                          6014
033116  001403                          1403
033120  005237  003104                  INC     ADJDIR
033124  004537  025330                  JSR     R5,ADJCYL
033130     002     000                  .BYTE   2,0             ;IN 2/0 OUTSIDE
```

```
                                                .WORD   170000
                                                .BYTE   2,0             ;OUT 2/0 OUTSIDE
                                                .WORD   7400
                                                .BYTE   4,0             ;IN 2/0 INSIDE
                                                .WORD   170000
                                                .BYTE   4,0             ;OUT 2/0 INSIDE
                                                .WORD   7400
                                                .BYTE   1,200           ;IN 2/1 OUTSIDE
                                                .WORD   0
                                                .BYTE   1,100           ;OUT 2/1 OUTSIDE
                                                .WORD   0
                                                .BYTE   5,200           ;IN 2/1 INSIDE
                                                .WORD   0
                                                .BYTE   5,100           ;OUT 2/1 INSIDE
                                                .WORD   0
                                                .WORD   0               ;TERMINATOR
033172  004537  032174                          JSR     R5,UNLOAD       ;UNLOAD PACK ON THIRD DRIVE
033176  023727  003130  000003                  CMP     UUT,#3          ;CHECK FOR > 3 DRIVES
033204  001500                                  BEQ     1$              ;NO, GO TO 2ND DRIVE
033206  062704  000010                          ADD     #PAT+2,R4       ;UPDATE DRIVE BUFFER FOR 4TH DRIVE
033212  004537  032300                          JSR     R5,LOAD         ;LOAD PACK ON 4TH DRIVE
033216  004537  031526                          JSR     R5,SERNUM       ;CHECK PACK ON FOURTH DRIVE
033222  004537  024606                          JSR     R5,OVWPER       ;PERFORM R/W OVERWRITE
033226  001042                                  1042
033230  000421                                  421
033232  005237  003104                          INC     ADJDIR
033236  004537  025330                          JSR     R5,ADJCYL
033242      002     000                         .BYTE   2,0             ;IN 3/0 OUTSIDE
033244  000360                                  .WORD   360
033246      002     000                         .BYTE   2,0             ;OUT 3/0 OUTSIDE
033250  000017                                  .WORD   17
033252      004     000                         .BYTE   4,0             ;IN 3/0 INSIDE
033254  000360                                  .WORD   360
033256      004     000                         .BYTE   4,0             ;OUT 3/0 INSIDE
033260  000017                                  .WORD   17
033262      001     040                         .BYTE   1,40            ;IN 3/1 OUTSIDE
033264  000000                                  .WORD   0
033266      001     020                         .BYTE   1,20            ;OUT 3/1 OUTSIDE
033270  000000                                  .WORD   0
033272      005     040                         .BYTE   5,40            ;IN 3/1 INSIDE
033274  000000                                  .WORD   0
033276      005     020                         .BYTE   5,20            ;OUT 3/1 INSIDE
033300  000000                                  .WORD   0
033302      001     000                         .BYTE   1,0             ;IN 3/2 OUTSIDE
033304  100000                                  .WORD   100000
033306      001     000                         .BYTE   1,0             ;OUT 3/2 OUTSIDE
033310  040000                                  .WORD   40000
033312      005     000                         .BYTE   5,0             ;IN 3/2 INSIDE
033314  100000                                  .WORD   100000
033316      005     000                         .BYTE   5,0             ;OUT 3/2 INSIDE
033320  040000                                  .WORD   40000
033322  000000                                  .WORD   0               ;TERMINATOR
033324  004537  032174                          JSR     R5,UNLOAD       ;UNLOAD PACK FROM 4TH DRIVE
033330  162704  000010                          SUB     #PAT+2,R4       ;SET DRIVE BUFFER FOR 3RD DRIVE
033334  004537  032300                          JSR     R5,LOAD         ;LOAD PACK ON 3RD DRIVE
033340  004537  031526                          JSR     R5,SERNUM       ;CHECK FOR PACK SERIAL NUMBER
033344  004537  024606                          JSR     R5,OVWPER       ;PERFORM R/W OVERWRITE ON 3RD DRIVE
```

```
033350  020000                          20000
033352  010000                          10000
033354  004537  025330                  JSR     R5,ADJCYL
033360     001     000                  .BYTE   1,0             ;IN 2/3 OUTSIDE
033362  000200                          .WORD   200
033364     001     000                  .BYTE   1,0             ;OUT 2/3 OUTSIDE
033366  000100                          .WORD   100
033370     005     000                  .BYTE   5,0             ;IN 2/3 INSIDE
033372  000200                          .WORD   200
033374     005     000                  .BYTE   5,0             ;OUT 2/3 INSIDE
033376  000100                          .WORD   100
033400  000000                          .WORD   0               ;TERMINATOR
033402  004537  032174                  JSR     R5,UNLOAD       ;UNLOAD PACK FROM 3RD DRIVE
033406  162704  000010          1$:     SUB     #PAT+2,R4       ;SET DRIVE BUFFER FOR 2ND DRIVE
033412  004537  032300                  JSR     R5,LOAD         ;LOAD PACK ON THIRD DRIVE
033416  004537  031526                  JSR     R5,SERNUM       ;CHECK PACK SERIAL NUMBER
033422  004537  024606                  JSR     R5,OVWPER       ;PERFORM R/W OVERWRITE ON 2ND DRIVE
033426  004040                          4040
033430  002020                          2020
033432  004537  025330                  JSR     R5,ADJCYL
033436     001     000                  .BYTE   1,0             ;IN 1/2 OUTSIDE
033440  020000                          .WORD   20000
033442     001     000                  .BYTE   1,0             ;OUT 1/2 OUTSIDE
033444  010000                          .WORD   10000
033446     005     000                  .BYTE   5,0             ;IN 1/2 INSIDE
033450  020000                          .WORD   20000
033452     005     000                  .BYTE   5,0             ;OUT 1/2 INSIDE
033454  010000                          .WORD   10000
033456     001     000                  .BYTE   1,0             ;IN 1/3 OUTSIDE
033460  000040                          .WORD   40
033462     001     000                  .BYTE   1,0             ;OUT 1/3 OUTSIDE
033464  000020                          .WORD   20
033466     005     000                  .BYTE   5,0             ;IN 1/3 INSIDE
033470  000040                          .WORD   40
033472     005     000                  .BYTE   5,0             ;OUT 1/3 INSIDE
033474  000020                          .WORD   20
033476  000000                          .WORD   0               ;TERMINATOR
033500  004537  032174                  JSR     R5,UNLOAD       ;UNLOAD PACK FROM 2ND DRIVE
033504  162704  000010          2$:     SUB     #PAT+2,R4       ;SET DRIVE BUFFER FOR 1ST DRIVE
033510  004537  032300                  JSR     R5,LOAD         ;LOAD PACK INTO FIRST DRIVE UNIT
033514  004537  031526                  JSR     R5,SERNUM       ;CHECK SERIAL NUMBER
033520  004537  024606                  JSR     R5,OVWPER       ;PERFORM R/W OVERWRITE
033524  001042                          1042
033526  000421                          421
033530  004537  025330                  JSR     R5,ADJCYL
033534     001     010                  .BYTE   1,10            ;IN 0/1 OUTSIDE
033536  000000                          .WORD   0
033540     001     004                  .BYTE   1,4             ;OUT 0/1 OUTSIDE
033542  000000                          .WORD   0
033544     005     010                  .BYTE   5,10            ;IN 0/1 INSIDE
033546  000000                          .WORD   0
033550     005     004                  .BYTE   5,4             ;OUT 0/1 INSIDE
033552  000000                          .WORD   0
033554     001     000                  .BYTE   1,0             ;IN 0/2 OUTSIDE
033556  004000                          .WORD   4000
033560     001     000                  .BYTE   1,0             ;OUT 0/2 OUTSIDE
033562  002000                          .WORD   2000
```

```
    172 033564     005      000                  .BYTE   5,0              ;IN 0/2 INSIDE
    173 033566  004000                          .WORD   4000
    174 033570     005      000                  .BYTE   5,0              ;OUT 0/2 INSIDE
    175 033572  002000                          .WORD   2000
    176 033574     001      000                  .BYTE   1,0              ;IN 0/3 OUTSIDE
    177 033576  000010                          .WORD   10
    178 033600     001      000                  .BYTE   1,0              ;OUT 0/3 OUTSIDE
    179 033602  000004                          .WORD   4
    180 033604     005      000                  .BYTE   5,0              ;IN 0/3 INSIDE
    181 033606  000010                          .WORD   10
    182 033610     005      000                  .BYTE   5,0              ;OUT 0/3 INSIDE
    183 033612  000004                          .WORD   4
    184 033614  000000                          .WORD   0                ;TERMINATOR
    185 033616  004537  032174                  JSR     R5,UNLOAD        ;UNLOAD PACK
    186 033622                                  PRINTF  #ENDPAS          ;END OF PASS
        033622  012746  022412                  MOV     #ENDPAS,-(SP)
        033626  012746  000001                  MOV     #1,-(SP)
        033632  010600                          MOV     SP,R0
        033634  104417                          TRAP    C$PNTF
        033636  062706  000004                  ADD     #4,SP
    187
    188 033642  000137  024266                  JMP     CMPENA           ;RETURN TO SUPERVISOR
    189
    190
    191 033646                          ENDTST
        033646                          L10014:
        033646  104401                          TRAP    C$ETST
    192 033650                          ENDMOD
    193
    194 033650                          BGNMOD  HRDPRM
    195 033650                                  BGNHRD
        033650  000025                          .WORD L10015-L$HARD/2
    196
    197 033652                                  GPRMA   CSRMSG,CSR,0,160000,177776,YES
        033652  000031                          .WORD   T$CODE
        033654  033724                          .WORD   CSRMSG
        033656  160000                          .WORD   T$LOLIM
        033660  177776                          .WORD   T$HILIM
    198
    199 033662                                  GPRMA   VECMSG,VECT,0,0,776,YES
        033662  001031                          .WORD   T$CODE
        033664  033762                          .WORD   VECMSG
        033666  000000                          .WORD   T$LOLIM
        033670  000776                          .WORD   T$HILIM
    200
    201 033672                                  GPRMD   DRMSG,DRBT,0,03400,0,7,YES
        033672  004032                          .WORD   T$CODE
        033674  033771                          .WORD   DRMSG
        033676  003400                          .WORD   03400
        033700  000000                          .WORD   T$LOLIM
        033702  000007                          .WORD   T$HILIM
    202
    203 033704                                  GPRML   DRTYPE,TYPDR,1,YES
        033704  003130                          .WORD   T$CODE
        033706  033740                          .WORD   DRTYPE
        033710  000001                          .WORD   1
    204
```

```
    205 033712                                      GPRMD   BRMSG,PRIOR,0,340,0,7,YES
        033712  002032                              .WORD   T$CODE
        033714  033777                              .WORD   BRMSG
        033716  000340                              .WORD   340
        033720  000000                              .WORD   T$LOLIM
        033722  000007                              .WORD   T$HILIM
    206
    207 033724                                      ENDHRD
                                                    .EVEN
        033724                              L10015:
    208
    209 033724      102     125     123     CSRMSG: .ASCIZ  /BUS ADDRESS/
        033727      040     101     104
        033732      104     122     105
        033735      123     123     000
    210 033740      104     122     111     DRTYPE: .ASCIZ  /DRIVE TYPE = RL01/
        033743      126     105     040
        033746      124     131     120
        033751      105     040     075
        033754      040     122     114
        033757      060     061     000
    211 033762      126     105     103     VECMSG: .ASCIZ  /VECTOR/
        033765      124     117     122
        033770      000
    212 033771      104     122     111     DRMSG:  .ASCIZ  /DRIVE/
        033774      126     105     000
    213 033777      102     122     040     BRMSG:  .ASCIZ  /BR LEVEL/
        034002      114     105     126
        034005      105     114     000
    214
    215                                     .EVEN
    216
    217 034010                              ENDMOD
    218
    219 034010                              LASTAD
                                                    .EVEN
        034010  000000                              .WORD   0
        034012  000000                              .WORD   0
        034014                              L$LAST::
    220
    221         000001                      .END
```

SYMBOL TABLE

```
ADJCYL  025330          CRDY   = 000200          C$SVEC= 000037          FRM12   021646          G$OFSI= 000376
ADJDIR  003104          CRSET  = 000002          C$TPRI= 000013          FRM13   021725          G$PRMA= 000001
ADJFLG  003102          CS     = 000000          DA     = 000004          FRM14   022010          G$PRMD= 000002
ADJLC2  003220          CSR    = 000000          DCKER   017455          FRM15   022037          G$PRML= 000000
ADJLC3  003222          CSRMSG  033724          DCRC   = 004000          FRM16   022076          G$RADA= 000140
ADJLC4  003224          CYL     003122          DERR   = 040000          FRM17   022163          G$RADB= 000000
ADJLOC  003100          C$AU   = 000052          DIAGMC= 000000          FRM18   022217          G$RADD= 000040
ADJTRK  003214          C$AUTO= 000061          DIRC    003116          FRM19   022303          G$RADL= 000120
ADJTXT  020210          C$BRK  = 000022          DLT    = 010000          FRM2    021067          G$RADO= 000020
ADJUUT  003216          C$BSEG= 000004          DRBT   = 000010          FRM20   022341          G$XFER= 000004
ADR    = 000020 G       C$BSUB= 000002          DRBUF   017240          FRM3    021167          G$YES  = 000010
ASSEMB= 000010          C$CEFG= 000045          DRDY   = 000001          FRM4    021246          HCRC   = 004000
AUTOCO  024312 G        C$CLCK= 000062          DRMSG   033771          FRM5    021307          HDRFND  002234
BA     = 000002          C$CLEA= 000012          DRPCOD  024322 G        FRM6    021356          HEAD   = 000100
BA16   = 000020          C$CLOS= 000035          DRST   = 000013          FRM7    021377          HEAD01  003114
BA17   = 000040          C$CLP1= 000006          DRSTAT  003106          FRM8    021437          HNF    = 010000
BBA     003202          C$CVEC= 000036          DRTYPE  033740          FRM9    021503          HOE    = 100000 G
BCS     003200          C$DCLN= 000044          DSB    = 000004          FRTTRK  003152          HPTCOD  022446 G
BDA     003204          C$DODU= 000051          DSPCOD  022462 G        FUNERR  020047          HRDPRM  033650 G
BDATA   003140          C$DRPT= 000024          EF.CON= 000036 G        FWD     020116          HRDWTS  032706 G
BIT0   = 000001 G       C$DU   = 000053          EF.NEW= 000035 G        F$AU   = 000015          HSFLG   003110
BIT00  = 000001 G       C$EDIT= 000003          EF.PWR= 000034 G        F$AUTO= 000020          IBE    = 010000 G
BIT01  = 000002 G       C$ERDF= 000055          EF.RES= 000037 G        F$BGN  = 000040          IDU    = 000040 G
BIT02  = 000004 G       C$ERHR= 000056          EF.STA= 000040 G        F$CLEA= 000007          IER    = 020000 G
BIT03  = 000010 G       C$ERRO= 000060          END     023020          F$DU   = 000016          INAWR   030454
BIT04  = 000020 G       C$ERSF= 000054          ENDBUF  017300          F$END  = 000041          INITCO  022466 G
BIT05  = 000040 G       C$ERSO= 000057          ENDPAS  022412          F$HARD= 000004          INITWR  017373
BIT06  = 000100 G       C$ESCA= 000010          ERFLG   003074          F$HW   = 000013          INN10   002356
BIT07  = 000200 G       C$ESEG= 000005          ERR    = 100000          F$INIT= 000006          INN11   002370
BIT08  = 000400 G       C$ESUB= 000003          ERRFND  017743          F$JMP  = 000050          INN20   002360
BIT09  = 001000 G       C$ETST= 000001          ERR1    020240 G        F$MOD  = 000000          INN21   002372
BIT1   = 000002 G       C$EXIT= 000032          ERR2    020276 G        F$MSG  = 000011          INN30   002362
BIT10  = 002000 G       C$GETB= 000026          ERR3    020336 G        F$PROT= 000021          INN31   002374
BIT11  = 004000 G       C$GETW= 000027          ERR4    020464 G        F$PWR  = 000017          INN40   002364
BIT12  = 010000 G       C$GMAN= 000043          ERR5    020644 G        F$RPT  = 000012          INN41   002376
BIT13  = 020000 G       C$GPHR= 000042          ERR6    020704 G        F$SEG  = 000003          INN50   002366
BIT14  = 040000 G       C$GPLO= 000030          EVL    = 000004 G       F$SOFT= 000005          INN51   002400
BIT15  = 100000 G       C$GPRI= 000040          EXIT    024310          F$SRV  = 000010          INTEN  = 000100
BIT2   = 000004 G       C$INIT= 000011          E$END  = 002100          F$SUB  = 000002          ISR    = 000100 G
BIT3   = 000010 G       C$INLP= 000020          E$LOAD= 000035          F$SW   = 000014          IXE    = 004000 G
BIT4   = 000020 G       C$MANI= 000050          E.BA    003166          F$TEST= 000001          I$AU   = 000041
BIT5   = 000040 G       C$MEM  = 000031          E.CS    003164          GDATA   003136          I$AUTO= 000041
BIT6   = 000100 G       C$MSG  = 000023          E.DA    003170          GLBDAT  002234 G        I$CLN  = 000041
BIT7   = 000200 G       C$OPEN= 000034          E.MP    003172          GLBEQA  002234 G        I$DU   = 000041
BIT8   = 000400 G       C$PNTB= 000014          E.MP1   003174          GLBERR  020240 G        I$HRD  = 000041
BIT9   = 001000 G       C$PNTF= 000017          E.MP2   003176          GLBSUB  024326 G        I$INIT= 000041
BMP     003206          C$PNTS= 000016          FADJ    003064          GLBTXT  017302 G        I$MOD  = 000041
BOE    = 000400 G       C$PNTX= 000015          FEW     017473          GSBIT  = 000003          I$MSG  = 000041
BRMSG   033777          C$QIO  = 000377          FIXCYL  031472          GSTAT  = 000004          I$PROT= 000040
BSEXIT  030540          C$RDBU= 000007          FNDBSC  031402          G$CNTO= 000200          I$PTAB= 000041
BSVWR   030356          C$REFG= 000047          FNDDRV  027140          G$DELM= 000372          I$PWR  = 000041
BUF     003232          C$RESE= 000033          FNDTRK  031230          G$DISP= 000003          I$RPT  = 000041
CHECK   030566          C$REVI= 000003          FORSK   003126          G$EXCP= 000400          I$SEG  = 000041
CLNCOD  024316 G        C$RFLA= 000021          FOWR    003062          G$HILI= 000002          I$SETU= 000041
CMPENA  024266          C$RPT  = 000025          FRM1    020772          G$LOLI= 000001          I$SRV  = 000041
CNTTOT  017346          C$SEFG= 000046          FRM10   021545          G$NO   = 000000          I$SUB  = 000041
COMPAT  032706          C$SPRI= 000041          FRM11   021611          G$OFFS= 000400          I$TST  = 000041
```

```
J$JMP = 000167          L$SPC   002056 G        OQU40   002270          REV     020126          T$HILI= 000007
LDFUNC  032404          L$SPCP  002020 G        OQU41   002302          REVSK   003124          T$LAST= 000001
LOAD    032300          L$SPTP  002024 G        OQU50   002272          RSADJS  026732          T$LOLI= 000000
LOE   = 040000 G        L$STA   002030 G        OQU51   002304          SECBUF  002442          T$LSYM= 010000
LOT   = 000010 G        L$TEST  002114 G        OSECT   003112          SECLST  002402          T$LTNO= 000001
LSTCLR  003070          L$TIML  002014 G        OUT10   002236          SECT    003132          T$NEST= 177777
LSTDRV  003134          L$UNIT  002012 G        OUT11   002250          SECWRD  003144          T$NS0 = 000000
LSTTRK  003150          L10000  020274          OUT20   002240          SEEK  = 000006          T$NS1 = 000004
L$ACP   002110 G        L10001  020334          OUT21   002252          SERNM1  003210          T$PTNU= 000000
L$APT   002036 G        L10002  020462          OUT30   002242          SERNM2  003212          T$SAVL= 177777
L$AUT   002070 G        L10003  020642          OUT31   002254          SERNUM  031526          T$SEGL= 177777
L$AUTO  024312 G        L10004  020702          OUT40   002244          SETLST  027100          T$SUBN= 000000
L$CCP   002106 G        L10005  020770          OUT41   002256          SETUP   023072          T$TAGL= 177777
L$CLEA  024316 G        L10007  022462          OUT50   002246          SIGN  = 000004          T$TAGN= 010016
L$CO    002032 G        L10010  024310          OUT51   002260          SKCYL   026254          T$TEMP= 000000
L$DEPO  002011 G        L10011  024314          OVMES   017642          SKER    020077          T$TEST= 000001
L$DESC  002122 G        L10012  024320          OVWER   020003          SKHS  = 000020          T$TSIM= 177777
L$DESP  002076 G        L10013  024324          OVWPER  024606          STFLG   003076          T$TSTS= 000001
L$DEVP  002060 G        L10014  033646          OVWTRK  003022          STSEC   003230          T$$AUT= 010011
L$DISP  022464 G        L10015  033724          O$APTS= 000000          STSEC1  003226          T$$CLE= 010012
L$DLY   002116 G        MANY    017532          O$AU  = 000000          SURF    003120          T$$DU = 010013
L$DTP   002040 G        MDHEDR  002000 G        O$BGNR= 000000          SURFAC  003156          T$$HAR= 010015
L$DTYP  002034 G        MERGE   031002          O$BGNS= 000000          SVCGBL= 000000          T$$HW = 010007
L$DU    024322 G        MID10   002306          O$DU  = 000000          SVCINS= 000000          T$$INI= 010010
L$DUT   002072 G        MID11   002320          O$ERRT= 000000          SVCSUB= 177777          T$$MSG= 010005
L$DVTY  002222 G        MID20   002310          O$GNSW= 000000          SVCTAG= 000000          T$$PRO= 010006
L$EF    002052 G        MID21   002322          O$POIN= 000001          SVCTST= 177777          T$$TES= 010014
L$ENVI  002044 G        MID30   002312          O$SETU= 000000          S$LSYM= 010000          T.DRIV  003060
L$ETP   002102 G        MID31   002324          PAT   = 000006          TEM     003056          T1      032706 G
L$EXP1  002046 G        MID40   002314          PATLST  003046          TEMP    003066          UAM   = 000200 G
L$EXP4  002064 G        MID41   002326          PNT   = 001000 G        TIME    024326          UNLOAD  032174
L$EXP5  002066 G        MID50   002316          PRI   = 002000 G        TQU10   002332          UUT     003130
L$HARD  033652 G        MID51   002330          PRIOR = 000004          TQU11   002344          VAJWR   030112
L$HIME  002120 G        MK    = 000001          PRI00 = 000000 G        TQU20   002334          VEC   = 000002
L$HPCP  002016 G        MP    = 000006          PRI01 = 000040 G        TQU21   002346          VECMSG  033762
L$HPTP  002022 G        NCHECK  030552          PRI02 = 000100 G        TQU30   002336          VECT  = 000002
L$HW    022450 G        NONE    017571          PRI03 = 000140 G        TQU31   002350          VEROD   027564
L$ICP   002104 G        NXM   = 020000          PRI04 = 000200 G        TQU40   002340          VEROW   027200
L$INIT  022466 G        OBUFF   017236          PRI05 = 000240 G        TQU41   002352          WCOUNT  003142
L$LADP  002026 G        OFFSET  003146          PRI06 = 000300 G        TQU50   002342          WRITE = 000012
L$LAST  034014 G        OPI   = 002000          PRI07 = 000340 G        TQU51   002354          WRIT1   020136
L$LOAD  002100 G        OPR001  017302          PRSTRK  003154          TRKCNT  003162          WRSEC   025070
L$LUN   002074 G        OPR002  017321          RDHDR = 000010          TRKFND  003160          XDELAY  017232
L$MREV  002050 G        OQU10   002262          READ  = 000014          TYPDR = 000006          XTIME   024452
L$NAME  002000 G        OQU11   002274          READ1   020163          T$ARGC= 000001          X$ALWA= 000000
L$PRIO  002042 G        OQU20   002264          REASON  003072          T$CODE= 002032          X$FALS= 000040
L$PROT  022440 G        OQU21   002276          RECER   020023          T$ERRN= 000322          X$OFFS= 000400
L$PRT   002112 G        OQU30   002266          RECMS   017675          T$EXCP= 000000          X$TRUE= 000020
L$REPP  002062 G        OQU31   002300          REGDMP  026522          T$GMAN= 000000          YDELAY  017234
L$REV   002010 G

. ABS.  034014      000
        000000      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  28672 WORDS  ( 112 PAGES)
DYNAMIC MEMORY AVAILABLE FOR  71 PAGES
,A:CZRLLC/C=SVC33.SRC/P:1,A:CZRLLC
```

CROSS REFERENCE TABLE (CREF V04.00)

CROSS REFERENCE TABLE (CREF V04.00)

```
C$BSEG   1-22#
C$BSUB   1-22#
C$CEFG   1-22#
C$CLCK   1-22#
C$CLEA   1-22#    7-16
C$CLOS   1-22#
C$CLP1   1-22#
C$CVEC   1-22#
C$DCLN   1-22#    6-318
C$DODU   1-22#    6-314
C$DRPT   1-22#
C$DU     1-22#    7-22
C$EDIT   1-22#    1-29
C$ERDF   1-22#   12-54    12-57    16-39    17-35    26-31
C$ERHR   1-22#    6-115    6-130    6-144    6-159    6-182    6-205    6-229    6-252    6-276    6-299    10-48    18-44    20-40
        20-42
C$ERRO   1-22#
C$ERSF   1-22#    6-12     6-18     6-311
C$ERSO   1-22#   10-45    18-41
C$ESCA   1-22#
C$ESEG   1-22#
C$ESUB   1-22#
C$ETST   1-22#   27-191
C$EXIT   1-22#
C$GETB   1-22#
C$GETW   1-22#
C$GMAN   1-22#    6-47     6-56     6-92
C$GPHR   1-22#    6-27     6-41
C$GPLO   1-22#
C$GPRI   1-22#
C$INIT   1-22#    6-321
C$INLP   1-22#
C$MANI   1-22#
C$MEM    1-22#
C$MSG    1-22#    4-43     4-48     4-57     4-67     4-72     4-78
C$OPEN   1-22#
C$PNTB   1-22#    4-41     4-46     4-51     4-53     4-55     4-61     4-63     4-64     4-65     4-70     4-75     4-77     13-4
        13-6     13-17
C$PNTF   1-22#    6-45     6-54     6-71     6-91     16-46    16-53    17-44    17-50    21-35    24-70    24-74    25-5     25-22
        27-186
C$PNTS   1-22#
C$PNTX   1-22#
C$QIO    1-22#
C$RDBU   1-22#    8-15     8-31
C$REFG   1-22#    6-69
C$RESE   1-22#    1-22#
C$REVI   1-22#    1-29
C$RFLA   1-22#
C$RPT    1-22#
C$SEFG   1-22#
C$SPRI   1-22#    6-7
C$SVEC   1-22#
C$TPRI   1-22#
CHECK    19-24    19-38    20-6#
CLNCOD    7-11#
CMPENA    6-13     6-19     6-50     6-59     6-312#   27-188
```

CROSS REFERENCE TABLE (CREF V04.00)

```
CNTTOT   4-13#   26-31
COMPAT  27-9#
CRDY     1-67#
CRSET    1-70#
CS       1-44#   13-13*   13-14    26-16*   26-17    26-18*   26-19    26-25    26-34
CSR      1-51#    1-85#    3-75     3-75     3-75     3-75     4-46     4-51     4-55     4-61     4-70     4-75     6-28*    13-9
        16-31    16-37    17-25    25-5     25-9*    25-10    25-22    25-26*   25-27    26-8     27-197   27-197   27-197
CSRMSG  27-197   27-209#
CYL      3-20#    4-53     4-63     9-20*    9-26*    9-27*    9-35*    9-38*    9-40*    9-42*    9-44*    10-18    11-33*   11-42*
        11-85*   11-88*   11-92*   12-15    12-44    18-12    19-12    19-23*   19-25*   19-37*   19-40*   19-43*   20-9
DA       1-46#   13-10*   26-10*   26-27    26-36
DCKER    4-154
DCRC     1-63#
DERR     1-60#
DIAGMC   1-22     1-22
DIRC     3-18#    4-65     9-32*    9-50*    11-45*   11-95*
DLT      1-62#
DRBT     1-89#    6-52    27-201   27-201   27-201
DRBUF    3-62#    6-23     6-66
DRDY     1-57#   25-10    25-27
DRMSG   27-201   27-212#
DRPCOD   7-19#
DRST     1-76#
DRSTAT   3-14#   13-16*   13-17
DRTYPE  27-203   27-210#
DSB      1-53#    3-75     3-75     3-75     3-75     4-46     4-51     4-55     4-61     4-70     4-75     6-32*    6-52     13-12
        16-32    16-38    17-26    25-5     25-8     25-22    25-25    26-13
DSPCOD   5-25#
E$END    1-22#
E$LOAD   1-22#    1-29
E.BA     3-38#   13-6     26-26*   26-35*
E.CS     3-37#   13-6     13-7     16-25    17-32    26-25*   26-34*   26-41
E.DA     3-39#   13-6     26-27*   26-36*
E.MP     3-40#    4-77    12-20*   12-21    12-41*   12-51    13-6     24-10    24-32    26-28*   26-37*
E.MF1    3-41#   26-29*   26-38*
E.MP2    3-42#   26-30*   26-39*
EF.CON   1-42#
EF.NEW   1-42#
EF.PWR   1-42#    6-69
EF.RES   1-42#
EF.STA   1-42#
END      6-26     6-65#
ENDBUF   3-80#
ENDPAS   4-108#  27-186
ERFLG    3-9#    10-41    12-10    12-34    12-39    18-37    20-34    21-25    24-50    26-7*    26-43*
ERR      1-59#
ERR1     4-39#    6-115    6-130    6-144    6-159    6-182    6-205    6-229    6-252    6-276    6-299
ERR2     4-45#   10-45    10-48    18-41    18-44    20-42
ERR3     4-50#   20-40
ERR4     4-59#   16-39    17-35
ERR5     4-69#   12-57    26-31
ERR6     4-74#   12-54
ERRFND   4-21#    6-115    6-130    6-144    6-159    6-182    6-205    6-229    6-252    6-276    6-299
EVL      1-42#
EXIT     6-308    6-320#
F$AU     1-22#
```

CROSS REFERENCE TABLE (CREF V04.00)

F$AUTO	1-22#	7-3	7-7											
F$BGN	1-22#	1-28	1-30	1-40	1-91	2-4	3-81	4-2	4-33	4-37	4-39	4-45	4-50	4-59
	4-69	4-74	4-115	5-4	5-13	5-22	5-25	5-29	6-3	6-5	6-322	7-2	7-3	7-8
	7-11	7-12	7-17	7-19	7-20	7-23	8-3	26-50	27-1	27-2	27-191	27-192	27-194	27-195
	27-217													
F$CLEA	1-22#	7-12	7-16											
F$DU	1-22#	7-20	7-22											
F$END	1-22	1-22	1-22	1-22	1-22	1-22	1-22	1-22	1-22	1-22	1-22	1-22	1-22	1-22
	1-22	1-22	1-22#	1-28	1-30	1-40	1-91	2-4	3-81	4-2	4-33	4-37	4-43	4-48
	4-57	4-67	4-72	4-78	4-115	5-13	5-22	5-25	5-29	6-3	6-321	6-322	7-2	7-7
	7-8	7-11	7-16	7-17	7-19	7-22	7-23	8-3	26-50	27-1	27-2	27-2	27-2	27-191
	27-191	27-192	27-194	27-207	27-217									
F$HARD	1-22#	27-195	27-207											
F$HW	1-22#	5-14	5-20											
F$INIT	1-22#	6-5	6-321											
F$JMP	1-22#													
F$MOD	1-22#	1-28	1-30	1-40	1-91	2-4	3-81	4-2	4-33	4-37	4-115	5-13	5-22	5-25
	5-29	6-3	6-322	7-2	7-8	7-11	7-17	7-19	7-23	8-3	26-50	27-1	27-192	27-194
	27-217													
F$MSG	1-22#	4-39	4-43	4-45	4-48	4-50	4-57	4-59	4-67	4-69	4-72	4-74	4-78	
F$PROT	1-22#	5-4	5-10											
F$PWR	1-22#													
F$RPT	1-22#													
F$SEG	1-22#													
F$SOFT	1-22#													
F$SRV	1-22#													
F$SUB	1-22#													
F$SW	1-22#													
F$TEST	1-22#	27-2	27-191											
FADJ	3-5#	6-67*	11-140	11-142*	27-19*									
FEW	4-16#	6-12												
FIXCYL	6-119	6-134	6-148	6-163	6-186	6-209	6-233	6-256	6-280	6-303	23-28#			
FNDBSC	22-13	23-4#												
FNDDRV	15-24#	16-36	17-24											
FNDTRK	6-105	6-121	6-135	6-150	6-165	6-174	6-187	6-196	6-211	6-220	6-234	6-243	6-258	6-267
	6-281	6-290	22-1#											
FORSK	3-22#	9-12*	9-29											
FOWR	3-4#	6-68*	16-28	17-13	27-14*	27-18*								
FRM1	4-88#	25-5												
FRM10	4-41	4-97#												
FRM11	4-98#	6-71												
FRM12	4-99#	13-4												
FRM13	4-100#	13-6												
FRM14	4-101#	13-17												
FRM15	4-102#	21-35	24-74											
FRM16	4-55	4-103#												
FRM17	4-77	4-104#												
FRM18	4-105#	6-91												
FRM19	4-106#	6-45												
FRM2	4-89#	25-22												
FRM20	4-107#	6-54												
FRM3	4-90#	24-70												
FRM4	4-46	4-51	4-61	4-70	4-75	4-91#								
FRM5	4-53	4-63	4-92#											
FRM6	4-64	4-93#												
FRM7	4-65	4-94#												

CROSS REFERENCE TABLE (CREF V04.00)

CROSS REFERENCE TABLE (CREF V04.00)

```
I$PWR     1-22#
I$RPT     1-22#
I$SEG     1-22#    27-2
I$SETU    1-22#
I$SRV     1-22#
I$SUB     1-22#    27-2
I$TST     1-22#    27-2      27-2#     27-191    27-191#   27-191#
IBE       1-42#
IDU       1-42#
IER       1-42#
INAWR    19-21     19-27#
INITCO    6-3#
INITWR    4-14#    16-30
INN10     2-51#     6-147
INN11     2-56#     6-162
INN20     2-52#
INN21     2-57#
INN30     2-53#     2-80
INN31     2-58#     2-85
INN40     2-54#
INN41     2-59#
INN50     2-55#
INN51     2-60#    11-134
INTEN     1-58#
ISR       1-42#
IXE       1-42#
J$JMP     1-22#
L$ACP     1-29#
L$APT     1-29#
L$AUT     1-29#
L$AUTO    1-29      7-3#
L$CCP     1-29#
L$CLEA    1-29      7-12#
L$CO      1-29#
L$DEPO    1-29#
L$DESC    1-29      1-32#
L$DESP    1-29#
L$DEVP    1-29#
L$DISP    1-29      5-27#
L$DLY     1-29#     8-13*     8-17      8-21*     8-22      8-27*     8-33*     8-34      8-38
L$DTP     1-29#
L$DTYP    1-29#
L$DU      7-20#
L$DUT     1-29#
L$DVTY    1-29      1-34#
L$EF      1-29#
L$ENVI    1-29#
L$ETP     1-29#
L$EXP1    1-29#
L$EXP4    1-29#
L$EXP5    1-29#
L$HARD    1-29     27-195    27-195#
L$HIME    1-29#
L$HPCP    1-29#
L$HPTP    1-29#
L$HW      1-29      5-14      5-14#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
L$ICP     1-29#
L$INIT    1-29      6-5#
L$LADP    1-29#
L$LAST    1-29     27-219#
L$LOAD    1-29#
L$LUN     1-29#
L$MREV    1-29#
L$NAME    1-29#
L$PRIO    1-29#
L$PROT    1-29      5-4#
L$PRT     1-29#
L$REPP    1-29#
L$REV     1-29#
L$SPC     1-29#
L$SPCP    1-29#
L$SPTP    1-29#
L$STA     1-29#
L$TEST    1-29#
L$TIML    1-29#
L$UNIT    1-29#     6-9       6-15      6-21      6-38      6-65      6-313
L10000    4-43#
L10001    4-48#
L10002    4-57#
L10003    4-67#
L10004    4-72#
L10005    4-78#
L10007    5-14      5-20#
L10010    6-321#
L10011    7-7#
L10012    7-16#
L10013    7-22#
L10014   27-191#
L10015   27-195    27-207#
LDFUNC    10-39     12-7      12-31     12-37     16-22     17-29     18-35     20-32     21-23     24-6      24-8      24-28     24-30     24-48
          26-4#
LOAD      6-96      24-72     25-22#    25-32     27-29     27-50     27-78     27-112    27-129    27-154
LOE       1-42#
LOT       1-42#
LSTCLR    3-7#      4-64      16-31*    16-37*    17-25*
LSTDRV    3-25#     4-64       4-64     16-32*    16-38*    17-26*
LSTTRK    3-31#     4-41      22-8*     22-27
MANY      4-17#     6-18
MDHEDR    1-28#
MERGE     6-98     21-12#
MID10     2-31#     6-185
MID11     2-36#     6-208
MID20     2-32#
MID21     2-37#
MID30     2-33#     2-78
MID31     2-38#     2-83
MID40     2-34#
MID41     2-39#
MID50     2-35#
MID51     2-40#
MK        1-78#    12-26
MP        1-47#    13-16     26-9*     26-28     26-29     26-30     26-37     26-38     26-39
```

CROSS REFERENCE TABLE (CREF V04.00)

```
PRSTRK   3-33#    22-11*   22-27    22-30*   23-10    23-28
RDHDR    1-73#    12-8     '2-38    24-9     24-31
READ     1-75#    16-23    17-30    18-36    20-33    21-24    24-49
READ1    4-29#    18-41    18-44    20-42
REASON   3-8#     4-64     16-30*   16-34*   17-34*
RECER    4-23#    17-35
RECMS    4-20#    17-34
REGDMP   4-47     4-52     4-62     4-71     4-76     13-4#
REV      4-27#    9-50     11-95
REVSK    3-21#    9-13*    9-47
RSADJS   14-10#   18-46    20-26
S$LSYM   1-22#    4-43#    4-48#    4-57#    4-67#    4-72#    4-78#    5-20#    6-47     6-47     6-47     6-47#    6-56     6-56
         6-56     6-56#    6-92     6-92     6-92     6-92#    6-321#   7-7#     7-16#    7-22#    27-191#  27-207#
SECBUF   2-70#    14-16    27-9
SECLST   2-66#    15-10    15-28
SECT     3-24#    4-53     4-63     16-12*   16-19    16-35    16-55    16-58*   16-59    16-61*   17-12*   17-19    17-23    17-52*
         17-53    17-55*   20-37*   20-39*
SECWRD   3-29#    16-41*   16-46    16-49*   16-50    17-38*   17-44    17-47*   17-48
SEEK     1-72#    12-32    24-29
SERNM1   3-47#    6-89*    24-62    24-64*   24-66    24-70
SERNM2   3-48#    6-90*    24-65*   24-68    24-70
SERNUM   6-97     24-5#    24-73    27-30    27-51    27-79    27-113   27-130   27-155
SETLST   15--5#   16-56
SETUP    6-70     6-88#
SIGN     1-79#    12-25
SKCYL    9-25     9-28     9-41     9-45     11-40    11-44    11-90    11-94    12-6#    19-47    20-7
SKER     4-25#    12-54
SKHS     1-80#    12-29
STFLG    3-10#    6-88*
STSEC    3-55#    11-13*   11-21*   11-22    11-24*   11-50    11-53*   11-54    11-64    11-100
STSEC1   3-54#    11-64*   11-65*   11-66    11-69*   11-73    11-112
SURF     3-19#    4-53     4-63     9-21*    9-24*    9-43*    10-22    11-32*   11-38*   12-27    12-48    18-18    19-13    19-42*
         20-14
SURFAC   3-34#    4-41     22-2*    23-13
SVCGBL   1-22#    1-28     1-28     1-28     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-32     1-32     1-32     1-34     1-34     1-34     1-40     1-40
         1-40     2-4      2-4      2-4      4-2      4-2      4-2      4-37     4-37     4-37     4-39     4-39     4-39     4-45
         4-45     4-45     4-50     4-50     4-50     4-59     4-59     4-59     4-69     4-69     4-69     4-74     4-74     4-74
         5-4      5-4      5-4      5-13     5-13     5-13     5-14     5-14     5-14     5-25     5-25     5-25     5-27     5-27
         5-27     6-3      6-3      6-3      6-5      6-5      6-5      7-2      7-2      7-2      7-3      7-3      7-3      7-11
         7-11     7-11     7-12     7-12     7-12     7-19     7-19     7-19     7-20     7-20     7-20     8-3      8-3      8-3
         27-1     27-1     27-1     27-194   27-194   27-194   27-195   27-195   27-195   27-219   27-219   27-219   27-219#
SVCINS   1-22#    1-23#    1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29
         1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-29     1-32     1-32
```

CROSS REFERENCE TABLE (CREF V04.00)

1-32	1-32	1-34	1-34	1-34	1-34	4-41	4-41	4-41	4-41	4-41	4-41	4-41	4-41
4-41	4-41	4-41	4-41	4-41	4-41	4-41	4-41	4-43	4-43	4-46	4-46	4-46	4-46
4-46	4-46	4-46	4-46	4-46	4-46	4-46	4-46	4-46	4-46	4-46	4-46	4-48	4-48
4-51	4-51	4-51	4-51	4-51	4-51	4-51	4-51	4-51	4-51	4-51	4-51	4-51	4-51
4-51	4-51	4-53	4-53	4-53	4-53	4-53	4-53	4-53	4-53	4-53	4-53	4-53	4-53
4-53	4-53	4-53	4-53	4-55	4-55	4-55	4-55	4-55	4-55	4-55	4-55	4-55	4-55
4-55	4-55	4-55	4-55	4-55	4-55	4-57	4-57	4-61	4-61	4-61	4-61	4-61	4-61
4-61	4-61	4-61	4-61	4-61	4-61	4-61	4-61	4-61	4-61	4-63	4-63	4-63	4-63
4-63	4-63	4-63	4-63	4-63	4-63	4-63	4-63	4-63	4-63	4-63	4-63	4-64	4-64
4-64	4-64	4-64	4-64	4-64	4-64	4-64	4-64	4-64	4-64	4-64	4-64	4-64	4-64
4-64	4-64	4-65	4-65	4-65	4-65	4-65	4-65	4-65	4-65	4-65	4-65	4-65	4-65
4-67	4-67	4-70	4-70	4-70	4-70	4-70	4-70	4-70	4-70	4-70	4-70	4-70	4-70
4-70	4-70	4-70	4-70	4-72	4-72	4-75	4-75	4-75	4-75	4-75	4-75	4-75	4-75
4-75	4-75	4-75	4-75	4-75	4-75	4-75	4-75	4-77	4-77	4-77	4-77	4-77	4-77
4-77	4-77	4-77	4-77	4-77	4-77	4-77	4-77	4-78	4-78	5-14	5-14	5-27	5-27
5-27	5-27	6-7	6-7	6-7	6-7	6-12	6-12	6-12	6-12	6-12	6-12	6-12	6-12
6-18	6-18	6-18	6-18	6-18	6-18	6-18	6-18	6-27	6-27	6-27	6-27	6-41	6-41
6-41	6-41	6-41	6-41	6-45	6-45	6-45	6-45	6-45	6-45	6-45	6-45	6-45	6-45
6-47	6-47	6-47	6-47	6-47	6-47	6-47	6-47	6-47	6-47	6-47	6-47	6-54	6-54
6-54	6-54	6-54	6-54	6-54	6-54	6-54	6-54	6-56	6-56	6-56	6-56	6-56	6-56
6-56	6-56	6-56	6-56	6-56	6-56	6-69	6-69	6-69	6-69	6-70	6-70	6-71	6-71
6-71	6-71	6-71	6-71	6-71	6-71	6-71	6-71	6-91	6-91	6-91	6-91	6-91	6-91
6-91	6-91	6-91	6-91	6-92	6-92	6-92	6-92	6-92	6-92	6-92	6-92	6-92	6-92
6-92	6-92	6-115	6-115	6-115	6-115	6-115	6-115	6-115	6-115	6-130	6-130	6-130	6-130
6-130	6-130	6-130	6-130	6-144	6-144	6-144	6-144	6-144	6-144	6-144	6-144	6-159	6-159
6-159	6-159	6-159	6-159	6-159	6-159	6-182	6-182	6-182	6-182	6-182	6-182	6-182	6-182
6-205	6-205	6-205	6-205	6-205	6-205	6-205	6-205	6-229	6-229	6-229	6-229	6-229	6-229
6-229	6-229	6-252	6-252	6-252	6-252	6-252	6-252	6-252	6-252	6-276	6-276	6-276	6-276
6-276	6-276	6-276	6-276	6-299	6-299	6-299	6-299	6-299	6-299	6-299	6-299	6-311	6-311
6-311	6-311	6-311	6-311	6-311	6-311	6-314	6-314	6-314	6-314	6-318	6-318	6-321	6-321
7-7	7-7	7-16	7-16	7-22	7-22	8-15	8-15	8-16	8-16	8-17	8-17	8-17	8-17
8-17	8-17	8-17	8-17	8-17	8-17	8-17	8-17	8-17	8-17	8-17	8-17	8-22	8-22
8-22	8-22	8-22	8-22	8-22	8-22	8-22	8-22	8-22	8-22	8-22	8-22	8-22	8-22
8-31	8-31	8-32	8-32	8-34	8-34	8-34	8-34	8-34	8-34	8-34	8-34	8-34	8-34
8-34	8-34	8-34	8-34	8-34	8-34	8-38	8-38	8-38	8-38	8-38	8-38	8-38	8-38
8-38	8-38	8-38	8-38	8-38	8-38	8-38	8-38	10-45	10-45	10-45	10-45	10-45	10-45
10-45	10-45	10-48	10-48	10-48	10-48	10-48	10-48	10-48	10-48	12-54	12-54	12-54	12-54
12-54	12-54	12-54	12-54	12-57	12-57	12-57	12-57	12-57	12-57	12-57	12-57	13-4	13-4
13-4	13-4	13-4	13-4	13-4	13-4	13-4	13-4	13-4	13-4	13-4	13-4	13-4	13-4
13-4	13-4	13-6	13-6	13-6	13-6	13-6	13-6	13-6	13-6	13-6	13-6	13-6	13-6
13-6	13-6	13-6	13-6	13-6	13-6	13-17	13-17	13-17	13-17	13-17	13-17	13-17	13-17
13-17	13-17	13-17	13-17	16-39	16-39	16-39	16-39	16-39	16-39	16-39	16-39	16-46	16-46
16-46	16-46	16-46	16-46	16-46	16-46	16-46	16-46	16-46	16-46	16-46	16-46	16-46	16-46
16-53	16-53	16-53	16-53	16-53	16-53	16-53	16-53	16-53	16-53	16-53	16-53	17-35	17-35
17-35	17-35	17-35	17-35	17-35	17-35	17-44	17-44	17-44	17-44	17-44	17-44	17-44	17-44
17-44	17-44	17-44	17-44	17-44	17-44	17-44	17-44	17-50	17-50	17-50	17-50	17-50	17-50
17-50	17-50	17-50	17-50	17-50	17-50	18-41	18-41	18-41	18-41	18-41	18-41	18-41	18-41
18-44	18-44	18-44	18-44	18-44	18-44	18-44	18-44	20-40	20-40	20-40	20-40	20-40	20-40
20-40	20-40	20-42	20-42	20-42	20-42	20-42	20-42	20-42	20-42	21-35	21-35	21-35	21-35
21-35	21-35	21-35	21-35	21-35	21-35	21-36	21-36	24-70	24-70	24-70	24-70	24-70	24-70
24-70	24-70	24-70	24-70	24-70	24-70	24-70	24-70	24-70	24-70	24-70	24-70	24-74	24-74
24-74	24-74	24-74	24-74	24-74	24-74	24-74	24-74	24-75	24-75	25-5	25-5	25-5	25-5
25-5	25-5	25-5	25-5	25-5	25-5	25-5	25-5	25-5	25-5	25-5	25-5	25-22	25-22
25-22	25-22	25-22	25-22	25-22	25-22	25-22	25-22	25-22	25-22	25-22	25-22	25-22	25-22
26-31	26-31	26-31	26-31	26-31	26-31	26-31	26-31	27-186	27-186	27-186	27-186	27-186	27-186
27-186	27-186	27-186	27-186	27-191	27-191	27-195	27-195	27-197	27-197	27-197	27-197	27-197	27-197

CROSS REFERENCE TABLE (CREF V04.00)

	27-197	27-197	27-199	27-199	27-199	27-199	27-199	27-199	27-199	27-199	27-201	27-201	27-201	27-201
	27-201	27-201	27-201	27-201	27-201	27-201	27-203	27-203	27-203	27-203	27-203	27-203	27-205	27-205
	27-205	27-205	27-205	27-205	27-205	27-205	27-205	27-205	27-207	27-207	27-219	27-219	27-219	27-219
	27-219	27-219												
SVCSUB	1-22#													
SVCTAG	1-22#	1-24#	4-43	4-43	4-43	4-48	4-48	4-48	4-57	4-57	4-57	4-67	4-67	4-67
	4-72	4-72	4-72	4-78	4-78	4-78	5-20	5-20	5-20	6-47	6-47	6-47	6-56	6-56
	6-56	6-92	6-92	6-92	6-321	6-321	6-321	7-7	7-7	7-7	7-16	7-16	7-16	7-22
	7-22	7-22	27-191	27-191	27-191	27-207	27-207	27-207						
SVCTST	1-22#	27-2	27-2	27-2										
T$$AUT	7-3#	7-7												
T$$CLE	7-12#	7-16												
T$$DU	7-20#	7-22												
T$$HAR	27-195	27-195#	27-207											
T$$HW	5-14	5-14#	5-20											
T$$INI	6-5#	6-321												
T$$MSG	4-39#	4-43	4-45#	4-48	4-50#	4-57	4-59#	4-67	4-69#	4-72	4-74#	4-78		
T$$PRO	5-4#													
T$$TES	27-2#	27-191												
T$ARGC	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29#	1-29#	1-29#
	1-29#	1-29#	1-29#	4-41	4-41	4-41	4-41	4-41	4-41#	4-41#	4-41#	4-41#	4-46	4-46
	4-46	4-46	4-46#	4-46#	4-46#	4-51	4-51	4-51	4-51	4-51#	4-51#	4-51#	4-53	4-53
	4-53	4-53	4-53	4-53#	4-53#	4-53#	4-53#	4-55	4-55	4-55	4-55	4-55#	4-55#	4-55#
	4-61	4-61	4-61	4-61	4-61#	4-61#	4-61#	4-63	4-63	4-63	4-63	4-63	4-63#	4-63#
	4-63#	4-63#	4-64	4-64	4-64	4-64	4-64	4-64	4-64#	4-64#	4-64#	4-64#	4-64#	4-65
	4-65	4-65	4-65#	4-65#	4-70	4-70	4-70	4-70	4-70#	4-70#	4-70#	4-75	4-75	4-75
	4-75	4-75#	4-75#	4-75#	4-77	4-77	4-77	4-77	4-77#	4-77#	4-77#	6-45	6-45	6-45#
	6-54	6-54	6-54#	6-71	6-71	6-71#	6-91	6-91	6-91#	13-4	13-4	13-4	13-4	13-4
	13-4	13-4#	13-4#	13-4#	13-4#	13-4#	13-6	13-6	13-6	13-6	13-6	13-6	13-6#	13-6#
	13-6#	13-6#	13-6#	13-17	13-17	13-17	13-17#	13-17#	16-46	16-46	16-46	16-46	16-46	16-46#
	16-46#	16-46#	16-46#	16-53	16-53	16-53	16-53#	16-53#	17-44	17-44	17-44	17-44	17-44	17-44#
	17-44#	17-44#	17-44#	17-50	17-50	17-50	17-50#	17-50#	21-35	21-35	21-35#	24-70	24-70	24-70
	24-70	24-70	24-70	24-70#	24-70#	24-70#	24-70#	24-70#	24-74	24-74	24-74#	25-5	25-5	25-5
	25-5	25-5#	25-5#	25-5#	25-22	25-22	25-22	25-22	25-22#	25-22#	25-22#	27-186	27-186	27-186#
T$CODE	6-47	6-47	6-47	6-47#	6-47#	6-47#	6-56	6-56	6-56	6-56#	6-56#	6-56#	6-92	6-92
	6-92	6-92#	6-92#	6-92#	27-197	27-197	27-197	27-197#	27-197#	27-197#	27-199	27-199	27-199	27-199#
	27-199#	27-199#	27-201	27-201	27-201	27-201#	27-201#	27-201#	27-203	27-203	27-203	27-203#	27-203#	27-203#
	27-205	27-205	27-205	27-205#	27-205#	27-205#								
T$ERRN	1-22#	6-12	6-12#	6-18	6-18#	6-115	6-115#	6-130	6-130#	6-144	6-144#	6-159	6-159#	6-182
	6-182#	6-205	6-205#	6-229	6-229#	6-252	6-252#	6-276	6-276#	6-299	6-299#	6-311	6-311#	10-45
	10-45#	10-48	10-48#	12-54	12-54#	12-57	12-57#	16-39	16-39#	17-35	17-35#	18-41	18-41#	18-44
	18-44#	20-40	20-40#	20-42	20-42#	26-31	26-31#							
T$EXCP	27-197	27-197#	27-199	27-199#	27-201	27-201#	27-205	27-205#						
T$GMAN	1-22#													
T$HILI	27-197	27-197#	27-199	27-199#	27-201	27-201#	27-205	27-205#						
T$LAST	1-22#	27-219#												
T$LOLI	27-197	27-197#	27-199	27-199#	27-201	27-201#	27-205	27-205#						
T$LSYM	1-22	1-22#	4-43	4-48	4-57	4-67	4-72	4-78	5-20	6-321	7-7	7-16	7-22	27-191
	27-207													
T$LTNO	27-219#													
T$NEST	1-22#	1-28	1-28	1-28#	1-30	1-30	1-30	1-30#	1-40	1-40	1-40#	1-91	1-91	1-91
	1-91#	2-4	2-4	2-4#	3-81	3-81	3-81	3-81#	4-2	4-2	4-2#	4-33	4-33	4-33
	4-33#	4-37	4-37	4-37#	4-39	4-39	4-39#	4-43	4-43	4-43	4-43#	4-45	4-45	4-45#
	4-48	4-48	4-48	4-48#	4-50	4-50	4-50#	4-57	4-57	4-57	4-57#	4-59	4-59	4-59#
	4-67	4-67	4-67	4-67#	4-69	4-69	4-69#	4-72	4-72	4-72	4-72#	4-74	4-74	4-74#
	4-78	4-78	4-78	4-78#	4-115	4-115	4-115	4-115#	5-4	5-4	5-4#	5-10	5-10	5-10

CROSS REFERENCE TABLE (CREF V04.00)

CROSS REFERENCE TABLE (CREF V04.00)

	6-273	6-287	6-296	22-9*	22-21*					
TYPDR	1-88#	6-43	27-203	27-203	27-203					
UAM	1-42#									
UNLOAD	24-71	25-5#	25-15	27-27	27-45	27-74	27-110	27-127	27-152	27-185
UUT	3-23#	6-21*	6-25	6-38	6-62*	6-65*	11-150	27-46	27-75	
VAJWR	11-59	11-78	11-106	11-119	18-6#					
VEC	1-52#	3-75	3-75	3-75	3-75	6-29*				
VECMSG	27-199	27-211#								
VECT	1-86#	27-199	27-199	27-199						
VEROD	9-34	9-52	17-8#							
VEROW	9-33	9-51	16-9#							
WCOUNT	3-28#	16-40*	16-45*	16-53	17-37*	17-43*	17-50			
WRIT1	4-28#	10-45	10-48							
WRITE	1-74#	10-40								
WRSEC	9-30	9-48	10-8#	11-56	11-75	11-102	11-114			
X$ALWA	1-22#									
X$FALS	1-22#									
X$OFFS	1-22#									
X$TRUE	1-22#									
XDELAY	3-57#	8-14*	8-18*	8-23*	26-21*					
XTIME	8-27#	25-12	25-29							
YDELAY	3-58#	8-28*	8-29*	8-30*	8-35*	8-39*	25-12*	25-29*		

CROSS REFERENCE TABLE (CREF V04.00)

CROSS REFERENCE TABLE (CREF V04.00)

	4-67	4-67#	4-72	4-72#	4-78	4-78#	4-115	4-115#	5-10	5-10#	5-20	5-20#	5-22	5-22#
	5-29	5-29#	6-321	6-321#	6-322	6-322#	7-7	7-7#	7-8	7-8#	7-16	7-16#	7-17	7-17#
	7-22	7-22#	7-23	7-23#	26-50	26-50#	27-191	27-191#	27-192	27-192#	27-207	27-207#	27-217	27-217#
M$DEFA	6-47	6-47#	6-56	6-56#	6-92	6-92#	27-197	27-197#	27-199	27-199#	27-201	27-201#	27-203	27-203#
	27-205	27-205#												
M$ENDE	1-30#	1-91#	3-81#	4-33#	4-43#	4-48#	4-57#	4-67#	4-72#	4-78#	4-115#	5-20#	5-22#	5-29#
	6-321#	6-322#	7-7#	7-8#	7-16#	7-17#	7-22#	7-23#	26-50#	27-191#	27-192#	27-207#	27-217#	
M$ERRI	6-12	6-12#	6-18	6-18#	6-115	6-115#	6-130	6-130#	6-144	6-144#	6-159	6-159#	6-182	6-182#
	6-205	6-205#	6-229	6-229#	6-252	6-252#	6-276	6-276#	6-299	6-299#	6-311	6-311#	10-45	10-45#
	10-48	10-48#	12-54	12-54#	12-57	12-57#	16-39	16-39#	17-35	17-35#	18-41	18-41#	18-44	18-44#
	20-40	20-40#	20-42	20-42#	26-31	26-31#								
M$EXCP	27-197	27-197	27-197#	27-199	27-199	27-199#	27-201	27-201	27-201#	27-205	27-205	27-205#		
M$GEN	1-28	1-28#	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29
	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29
	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29#	1-29#
	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#
	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#
	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-32	1-32#	1-34	1-34#	1-40	1-40#
	2-4	2-4#	4-2	4-2#	4-37	4-37#	4-39	4-39#	4-43	4-43#	4-45	4-45#	4-48	4-48#
	4-50	4-50#	4-57	4-57#	4-59	4-59#	4-67	4-67#	4-69	4-69#	4-72	4-72#	4-74	4-74#
	4-78	4-78#	5-4	5-4#	5-13	5-13#	5-14	5-14#	5-14#	5-20	5-20#	5-25	5-25#	5-27
	5-27#	6-3	6-3#	6-5	6-5#	6-47	6-47#	6-56	6-56#	6-92	6-92#	6-321	6-321#	7-2
	7-2#	7-3	7-3#	7-7	7-7#	7-11	7-11#	7-12	7-12#	7-16	7-16#	7-19	7-19#	7-20
	7-20#	7-22	7-22#	8-3	8-3#	27-1	27-1#	27-2	27-2#	27-191	27-191#	27-194	27-194#	27-195
	27-195#	27-207	27-207#	27-219	27-219#									
M$GENB	6-47	6-47#	6-56	6-56#	6-92	6-92#								
M$GETS	1-30	1-30#	1-91	1-91#	3-81	3-81#	4-33	4-33#	4-43	4-43#	4-48	4-48#	4-57	4-57#
	4-67	4-67#	4-72	4-72#	4-78	4-78#	4-115	4-115#	5-10	5-10#	5-20	5-20#	5-22	5-22#
	5-29	5-29#	6-321	6-321#	6-322	6-322#	7-7	7-7#	7-8	7-8#	7-16	7-16#	7-17	7-17#
	7-22	7-22#	7-23	7-23#	26-50	26-50#	27-191	27-191#	27-192	27-192#	27-207	27-207#	27-217	27-217#
M$GNGB	1-28	1-28#	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29
	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29
	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29#	1-29#
	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#
	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#
	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-32	1-32#	1-34	1-34#	1-40	1-40#
	2-4	2-4#	4-2	4-2#	4-37	4-37#	4-39	4-39#	4-45	4-45#	4-50	4-50#	4-59	4-59#
	4-69	4-69#	4-74	4-74#	5-4	5-4#	5-13	5-13#	5-14	5-14#	5-25	5-25#	5-27	
	5-27#	6-3	6-3#	6-5	6-5#	7-2	7-2#	7-3	7-3#	7-11	7-11#	7-12	7-12#	7-19
	7-19#	7-20	7-20#	8-3	8-3#	27-1	27-1#	27-194	27-194#	27-195	27-195#	27-219	27-219#	
M$GNIN	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29
	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29
	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29	1-29
	1-29	1-29	1-29	1-29	1-29	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#
	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#
	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#
	1-29#	1-29#	1-29#	1-29#	1-29#	1-29#	1-32	1-32	1-32#	1-32#	1-34	1-34	1-34#	1-34#
	4-41	4-41	4-41	4-41	4-41	4-41	4-41	4-41	4-41#	4-41#	4-41#	4-41#	4-41#	4-41#
	4-41#	4-43	4-43#	4-46	4-46	4-46	4-46	4-46	4-46	4-46	4-46	4-46#	4-46#	4-46#
	4-46#	4-46#	4-46#	4-48	4-48#	4-51	4-51	4-51	4-51	4-51	4-51	4-51	4-51	4-51#
	4-51#	4-51#	4-51#	4-51#	4-51#	4-53	4-53	4-53	4-53	4-53	4-53	4-53	4-53	4-53#
	4-53#	4-53#	4-53#	4-53#	4-53#	4-53#	4-55	4-55	4-55	4-55	4-55	4-55	4-55	4-55
	4-55#	4-55#	4-55#	4-55#	4-55#	4-55#	4-57	4-57#	4-61	4-61	4-61	4-61	4-61	4-61
	4-61	4-61	4-61#	4-61#	4-61#	4-61#	4-61#	4-61#	4-63	4-63	4-63	4-63	4-63	4-63
	4-63	4-63	4-63#	4-63#	4-63#	4-63#	4-63#	4-63#	4-63#	4-64	4-64	4-64	4-64	4-64
	4-64	4-64	4-64	4-64	4-64#	4-64#	4-64#	4-64#	4-64#	4-64#	4-64#	4-64#	4-65	4-65

	5-13	5-13#	5-14	5-14	5-14#	5-14#	5-25	5-25#	6-3	6-3#	6-5	6-5	6-5#	6-5#
	6-7#	6-12#	6-18#	6-27#	6-41#	6-45#	6-47	6-47#	6-47#	6-54#	6-56	6-56#	6-56#	6-69#
	6-71#	6-91#	6-92	6-92#	6-92#	6-115#	6-130#	6-144#	6-159#	6-182#	6-205#	6-229#	6-252#	6-276#
	6-299#	6-311#	6-314#	6-318#	6-321#	7-2	7-2#	7-3	7-3	7-3#	7-3#	7-7#	7-11	7-11#
	7-12	7-12	7-12#	7-12#	7-16#	7-19	7-19#	7-20	7-20	7-20#	7-20#	7-22#	8-3	8-3#
	8-15#	8-31#	10-45#	10-48#	12-54#	12-57#	13-4#	13-6#	13-17#	16-39#	16-46#	16-53#	17-35#	17-44#
	17-50#	18-41#	18-44#	20-40#	20-42#	21-35#	21-36#	24-70#	24-74#	24-75#	25-5#	25-22#	26-31#	27-1
	27-1#	27-2	27-2	27-2	27-2#	27-2#	27-2#	27-186#	27-191#	27-194	27-194#	27-195	27-195	27-195#
	27-195#													
MSLDRO	6-7	6-7#	6-27	6-27#	6-41	6-41#	6-69	6-69#	6-314	6-314#				
MSMCHI	1-22	1-22#												
MSMCLO	1-22	1-22#												
MSPOP	1-30	1-30#	1-91	1-91#	3-81	3-81#	4-33	4-33#	4-43	4-43#	4-48	4-48#	4-57	4-57#
	4-67	4-67#	4-72	4-72#	4-78	4-78#	4-115	4-115#	5-10	5-10#	5-20	5-20#	5-22	5-22#
	5-29	5-29#	6-321	6-321#	6-322	6-322#	7-7	7-7#	7-8	7-8#	7-16	7-16#	7-17	7-17#
	7-22	7-22#	7-23	7-23#	26-50	26-50#	27-191	27-191#	27-192	27-192#	27-207	27-207#	27-217	27-217#
MSPRIN	4-41	4-41#	4-46	4-46#	4-51	4-51#	4-53	4-53#	4-55	4-55#	4-61	4-61#	4-63	4-63#
	4-64	4-64#	4-65	4-65#	4-70	4-70#	4-75	4-75#	4-77	4-77#	6-45	6-45#	6-54	6-54#
	6-71	6-71#	6-91	6-91#	13-4	13-4#	13-6	13-6#	13-17	13-17#	16-46	16-46#	16-53	16-53#
	17-44	17-44#	17-50	17-50#	21-35	21-35#	24-70	24-70#	24-74	24-74#	25-5	25-5#	25-22	25-22#
	27-186	27-186#												
MSPUSH	1-28	1-28#	1-40	1-40#	2-4	2-4#	4-2	4-2#	4-37	4-37#	4-39	4-39#	4-45	4-45#
	4-50	4-50#	4-59	4-59#	4-69	4-69#	4-74	4-74#	5-4	5-4#	5-13	5-13#	5-14	5-14#
	5-25	5-25#	6-3	6-3#	6-5	6-5#	7-2	7-2#	7-3	7-3#	7-11	7-11#	7-12	7-12#
	7-19	7-19#	7-20	7-20#	8-3	8-3#	27-1	27-1#	27-2	27-2#	27-194	27-194#	27-195	27-195#
MSPUT	4-41	4-41	4-41	4-41	4-41	4-41#	4-46	4-46	4-46	4-46	4-46#	4-51	4-51	4-51
	4-51	4-51#	4-53	4-53	4-53	4-53	4-53	4-53#	4-55	4-55	4-55	4-55	4-55#	4-61
	4-61	4-61	4-61	4-61#	4-63	4-63	4-63	4-63	4-63	4-63#	4-64	4-64	4-64	4-64
	4-64	4-64	4-64#	4-65	4-65	4-65	4-65#	4-70	4-70	4-70	4-70	4-70#	4-75	4-75
	4-75	4-75	4-75#	4-77	4-77	4-77	4-77	4-77#	6-45	6-45	6-45#	6-54	6-54	6-54#
	6-71	6-71	6-71#	6-91	6-91	6-91#	13-4	13-4	13-4	13-4	13-4	13-4	13-4#	13-6
	13-6	13-6	13-6	13-6	13-6	13-6#	13-17	13-17	13-17	13-17#	16-46	16-46	16-46	16-46
	16-46	16-46#	16-53	16-53	16-53	16-53#	17-44	17-44	17-44	17-44	17-44	17-44#	17-50	17-50
	17-50	17-50#	21-35	21-35	21-35#	24-70	24-70	24-70	24-70	24-70	24-70	24-70#	24-74	24-74
	24-74#	25-5	25-5	25-5	25-5	25-5#	25-22	25-22	25-22	25-22	25-22#	27-186	27-186	27-186#
MSPUT1	4-41	4-41	4-41	4-41	4-41	4-41#	4-41#	4-41#	4-41#	4-41#	4-46	4-46	4-46	4-46
	4-46#	4-46#	4-46#	4-46#	4-51	4-51	4-51	4-51	4-51#	4-51#	4-51#	4-51#	4-53	4-53
	4-53	4-53	4-53	4-53#	4-53#	4-53#	4-53#	4-53#	4-55	4-55	4-55	4-55	4-55#	4-55#
	4-55#	4-55#	4-61	4-61	4-61	4-61	4-61#	4-61#	4-61#	4-61#	4-63	4-63	4-63	4-63
	4-63	4-63#	4-63#	4-63#	4-63#	4-63#	4-64	4-64	4-64	4-64	4-64	4-64	4-64#	4-64#
	4-64#	4-64#	4-64#	4-64#	4-65	4-65	4-65	4-65#	4-65#	4-65#	4-70	4-70	4-70	4-70
	4-70#	4-70#	4-70#	4-70#	4-75	4-75	4-75	4-75	4-75#	4-75#	4-75#	4-75#	4-77	4-77
	4-77	4-77	4-77#	4-77#	4-77#	4-77#	6-45	6-45	6-45#	6-45#	6-54	6-54	6-54#	6-54#
	6-71	6-71	6-71#	6-71#	6-91	6-91	6-91#	6-91#	13-4	13-4	13-4	13-4	13-4	13-4
	13-4#	13-4#	13-4#	13-4#	13-4#	13-4#	13-6	13-6	13-6	13-6	13-6	13-6	13-6#	13-6#
	13-6#	13-6#	13-6#	13-6#	13-17	13-17	13-17	13-17#	13-17#	13-17#	16-46	16-46	16-46	16-46
	16-46	16-46#	16-46#	16-46#	16-46#	16-46#	16-53	16-53	16-53	16-53#	16-53#	16-53#	17-44	17-44
	17-44	17-44	17-44	17-44#	17-44#	17-44#	17-44#	17-44#	17-50	17-50	17-50	17-50#	17-50#	17-50#
	21-35	21-35	21-35#	21-35#	24-70	24-70	24-70	24-70	24-70	24-70	24-70#	24-70#	24-70#	24-70#
	24-70#	24-70#	24-74	24-74	24-74#	24-74#	25-5	25-5	25-5	25-5	25-5#	25-5#	25-5#	25-5#
	25-22	25-22	25-22	25-22	25-22#	25-22#	25-22#	25-22#	27-186	27-186	27-186#	27-186#		
MSRADI	6-47	6-47#	6-56	6-56#	6-92	6-92#	27-197	27-197#	27-199	27-199#	27-201	27-201#	27-203	27-203#
	27-205	27-205#												
MSRNRO	6-27	6-27#	6-41	6-41#										
MSSETS	1-28	1-28#	1-40	1-40#	2-4	2-4#	4-2	4-2#	4-37	4-37#	4-39	4-39#	4-45	4-45#
	4-50	4-50#	4-59	4-59#	4-69	4-69#	4-74	4-74#	5-4	5-4#	5-13	5-13#	5-14	5-14#

CROSS REFERENCE TABLE (CREF V04.00)

	5-25	5-25#	6-3	6-3#	6-5	6-5#	7-2	7-2#	7-3	7-3#	7-11	7-11#	7-12	7-12#
	7-19	7-19#	7-20	7-20#	8-3	8-3#	27-1	27-1#	27-2	27-2#	27-194	27-194#	27-195	27-195#
MSSVC	4-41	4-41#	4-43	4-43#	4-46	4-46#	4-48	4-48#	4-51	4-51#	4-53	4-53#	4-55	4-55#
	4-57	4-57#	4-61	4-61#	4-63	4-63#	4-64	4-64#	4-65	4-65#	4-67	4-67#	4-70	4-70#
	4-72	4-72#	4-75	4-75#	4-77	4-77#	4-78	4-78#	6-7	6-7#	6-12	6-18	6-27	6-27#
	6-41	6-41#	6-45	6-45#	6-47	6-47#	6-54	6-54#	6-56	6-56#	6-69	6-69#	6-71	6-71#
	6-91	6-91#	6-92	6-92#	6-115	6-130	6-144	6-159	6-182	6-205	6-229	6-252	6-276	6-299
	6-311	6-314	6-314#	6-318	6-318#	6-321	6-321#	7-7	7-7#	7-16	7-16#	7-22	7-22#	8-15
	8-15#	8-31	8-31#	10-45	10-48	12-54	12-57	13-4	13-4#	13-6	13-6#	13-17	13-17#	16-39
	16-46	16-46#	16-53	16-53#	17-35	17-44	17-44#	17-50	17-50#	18-41	18-44	20-40	20-42	21-35
	21-35#	21-36	21-36#	24-70	24-70#	24-74	24-74#	24-75	24-75#	25-5	25-5#	25-22	25-22#	26-31
	27-186	27-186#	27-191	27-191#										
MSTLAB	4-41#	4-43#	4-46#	4-48#	4-51#	4-53#	4-55#	4-57#	4-61#	4-63#	4-64#	4-65#	4-67#	4-70#
	4-72#	4-75#	4-77#	4-78#	6-7#	6-12#	6-18#	6-27#	6-41#	6-45#	6-47#	6-54#	6-56#	6-69#
	6-71#	6-91#	6-92#	6-115#	6-130#	6-144#	6-159#	6-182#	6-205#	6-229#	6-252#	6-276#	6-299#	6-311#
	6-314#	6-318#	6-321#	7-7#	7-16#	7-22#	8-15#	8-31#	10-45#	10-48#	12-54#	12-57#	13-4#	13-6#
	13-17#	16-39#	16-46#	16-53#	17-35#	17-44#	17-50#	18-41#	18-44#	20-40#	20-42#	21-35#	21-36#	24-70#
	24-74#	24-75#	25-5#	25-22#	26-31#	27-186#	27-191#							
MSTSTL	4-41	4-41#	4-43	4-43#	4-46	4-46#	4-48	4-48#	4-51	4-51#	4-53	4-53#	4-55	4-55#
	4-57	4-57#	4-61	4-61#	4-63	4-63#	4-64	4-64#	4-65	4-65#	4-67	4-67#	4-70	4-70#
	4-72	4-72#	4-75	4-75#	4-77	4-77#	4-78	4-78#	6-7	6-7#	6-12	6-12#	6-12#	6-18
	6-18#	6-18#	6-27	6-27#	6-41	6-41#	6-45	6-45#	6-47	6-47#	6-54	6-54#	6-56	6-56#
	6-69	6-69#	6-71	6-71#	6-91	6-91#	6-92	6-92#	6-115	6-115#	6-115#	6-130	6-130#	6-130#
	6-144	6-144#	6-144#	6-159	6-159#	6-159#	6-182	6-182#	6-182#	6-205	6-205#	6-205#	6-229	6-229#
	6-229#	6-252	6-252#	6-252#	6-276	6-276#	6-276#	6-299	6-299#	6-299#	6-311	6-311#	6-311#	6-314
	6-314#	6-318	6-318#	6-321	6-321#	7-7	7-7#	7-16	7-16#	7-22	7-22#	8-15	8-15#	8-31
	8-31#	10-45	10-45#	10-45#	10-48	10-48#	10-48#	12-54	12-54#	12-54#	12-57	12-57#	12-57#	13-4
	13-4#	13-6	13-6#	13-17	13-17#	16-39	16-39#	16-39#	16-46	16-46#	16-53	16-53#	17-35	17-35#
	17-35#	17-44	17-44#	17-50	17-50#	18-41	18-41#	18-41#	18-44	18-44#	18-44#	20-40	20-40#	20-40#
	20-42	20-42#	20-42#	21-35	21-35#	21-36	21-36#	24-70	24-70#	24-74	24-74#	24-75	24-75#	25-5
	25-5#	25-22	25-22#	26-31	26-31#	26-31#	27-186	27-186#	27-191	27-191#				
MSWORD	1-29	1-29#	5-27	5-27	5-27#	6-12	6-12	6-12	6-12#	6-18	6-18	6-18	6-18#	6-47
	6-47	6-47#	6-47#	6-56	6-56	6-56#	6-56#	6-92	6-92	6-92#	6-92#	6-115	6-115	6-115
	6-115#	6-130	6-130	6-130	6-130#	6-144	6-144	6-144	6-144#	6-159	6-159	6-159	6-159#	6-182
	6-182	6-182	6-182#	6-205	6-205	6-205	6-205#	6-229	6-229	6-229	6-229#	6-252	6-252	6-252
	6-252#	6-276	6-276	6-276	6-276#	6-299	6-299	6-299	6-299#	6-311	6-311	6-311	6-311#	10-45
	10-45	10-45	10-45#	10-48	10-48	10-48	10-48#	12-54	12-54	12-54	12-54#	12-57	12-57	12-57
	12-57#	16-39	16-39	16-39	16-39#	17-35	17-35	17-35	17-35#	18-41	18-41	18-41	18-41#	18-44
	18-44	18-44	18-44#	20-40	20-40	20-40	20-40#	20-42	20-42	20-42	20-42#	26-31	26-31	26-31
	26-31#	27-197	27-197#	27-199	27-199#	27-201	27-201#	27-203	27-203#	27-205	27-205#	27-219	27-219	
POINTE	1-26													
PRINTB	4-41	4-46	4-51	4-53	4-55	4-61	4-63	4-64	4-65	4-70	4-75	4-77	13-4	13-6
	13-17													
PRINTF	6-45	6-54	6-71	6-91	16-46	16-53	17-44	17-50	21-35	24-70	24-74	25-5	25-22	27-186
READBU	8-15	8-31												
READEF	6-69													
SETPRI	6-7													
SVC	1-4#	1-22												
WAITMS	1-15#	25-12	25-29											
WAITUS	1-10#	26-21												