RABO-** UDA & DISK DRV DIAG AH-S831A-MC OCT 1981
CZUDCAO FICHE 1 OF 2

RA80.** UDA & DISK DRV DIAG AH-S831A-MC
CZUDCA0 FICHE 2 OF 2

```
.REM &
```

IDENTIFICATION

PRODUCT CODE: AC-S830A-MC

PRODUCT NAME: CZUDCAO UDA AND DISK DRIVE DIAGNOSTIC

PRODUCT DATE: 10-JULY-81

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: DALE KECK

TABLE OF CONTENTS

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

THE UDA HOST RESIDENT DIAGNOSTIC CAUSES THE EXECUTION OF THE UDA AND DRIVE RESIDENT DIAGNOSTIC INDIVIDUAL PROGRAMS AND REPORTS THE RESULTS ON THE PDP-11 CONSOLE TERMINAL OR LINE PRINTER. EXERCISES ALSO PERFORMS TO VERIFY THAT:

1. THE UDA CAN PROPERLY COMMUNICATE WITH THE PDP-11 PROCESSOR AND CAN TRANSFER BLOCKS OF DATA TO AND FROM UNIBUS MEMORY.

2. THE UDA AND THE DISK DRIVES CAN COMMUNICATE AND TRANSFER DATA PROPERLY.

3. THE DISK DRIVES CAN FUNCTION PROPERLY AS DEFINED BY THE SDI.

4. THE DISK DRIVES CAN SEEK, READ AND WRITE TO ALL BLOCKS ON THE DISK AND ALLOW THE MEASUREMENT OF THE DRIVES' PERFORMANCE IN A RATIO OF ERRORS PER MILLION BITS TRANSFERRED.

THE UDA HOST RESIDENT DIAGNOSTIC CONSISTS OF ONE PDP-11 DIAGNOSTIC SUPERVISOR PROGRAM THAT RUNS IN THE PDP-11 PROCESSOR AND FOUR PROGRAMS THAT RUNS IN THE UDA'S BUFFER MEMORY THROUGH AN INTERPRETER CALLED THE 'DIAGNOSTIC MACHINE'' WHICH RESIDES IN THE UDA. THE PDP-11 PROGRAM MAINLY IS RESPONSIBLE FOR DOWNLINE LOADING THE 'DIAGNOSTIC MACHINE'' PROGRAMS INTO THE UDA AND STARTING THEIR EXECUTION. THE 'DIAGNOSTIC MACHINE'' PROGRAM CONTROLS THE TESTING FROM THAT POINT BY REQUESTING THE PDP-11 PROCESSOR TO SUPPLY INFORMATION, PRINT ERROR MESSAGES AND UPDATE STATISTICS. THE 'DIAGNOSTIC MACHINE'' PROGRAM INFORMS THE PDP-11 PROCESSOR WHEN A TEST IS COMPLETE.

THE NUMBER OF UDAS AND DISK DRIVES THAT CAN BE TESTED AT ONE TIME BY THIS DIAGNOSTIC PROGRAM IS LIMITED BY THE PDP-11 MEMORY SIZE. WITH THE MINIMUM MEMORY SIZE OF 28K WORDS, AT LEAST TWO UDAS WITH FOUR DISK DRIVES EACH ARE SELECTABLE. THE NUMBER OF UDAS AND DISK DRIVES SELECTABLE BY THE DIAGNOSTIC INCREASES WITH ADDITIONAL MEMORY, BUT THERE IS NO GOAL IMPLIED FOR THE MAXIMUM POSSIBLE CONFIGURATION.

THIS DIAGNOSTIC HAS BEEN WRITTEN FOR USE WITH THE DIAGNOSTIC RUNTIME SERVICES SOFTWARE (SUPERVISOR). THESE SERVICES PROVIDE THE INTERFACE TO THE OPERATOR AND TO THE SOFTWARE ENVIRONMENT. THIS PROGRAM CAN BE USED WITH XXDP+, ACT, APT, SLIDE AND PAPER TAPE. FOR A COMPLETE DESCRIPTION OF THE RUNTIME SERVICES, REFER TO THE XXDP+ USER'S MANUAL. THERE IS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES IN SECTION 2 OF THIS DOCUMENT.

1.2 SYSTEM REQUIREMENTS

THIS PROGRAM WILL BE DESIGNED USING THE PDP-11 SUPERVISOR REVISION C. RUN TIME ENVIRONMENTS ARE DETERMINED BY THE SUPERVISOR AND MAY CHANGE AS NEW VERSIONS OF THE SUPERVISOR ARE DEVELOPED. THE INITIAL VERSION WILL REQUIRE THE FOLLOWING:

PDP-11 PROCESSOR
28K WORDS OF MEMORY (MINIMUM)
XXDP+ LOAD MEDIA
ONE OR MORE UDA SUB-SYSTEMS
LINE CLOCK - EITHER TYPE L OR P

THE LINE CLOCK WILL BE USED FOR ALL TIMED LOOPS IN THE PROGRAM. THE DIAGNOSTIC WILL RUN ON A SYSTEM WITH NO CLOCK BUT WILL HANG WHENEVER AN EVENT FOR WHICH THE PROGRAM IS WAITING DOES NOT HAPPEN (I.E., A TIME-OUT ERROR MESSAGE WILL NOT RESULT).

2.0 OPERATING INSTRUCTIONS

THIS SECTION CONTAINS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES. FOR DETAILED INFORMATION, REFER TO THE XXDP+ USER'S MANUAL (CHQUS).

2.1 COMMANDS

THERE ARE ELEVEN LEGAL COMMANDS FOR THE DIAGNOSTIC RUNTIME SERVICES (SUPERVISOR). THIS SECTION LISTS THE COMMANDS AND GIVES A VERY BRIEF DESCRIPTION OF THEM. THE XXDP+ USER'S MANUAL HAS MORE DETAILS.

COMMAND	EFFECT
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT TEST THAT WAS INTERRUPTED (AFTER ^C)
PROCEED	CONTINUE FROM AN ERROR HALT
EXIT	RETURN TO XXDP+ MONITOR (XXDP+ OPERATION ONLY.)
ADD	ACTIVATE A UNIT FOR TESTING (ALL UNITS ARE CONSIDERED TO BE ACTIVE AT START TIME
DROP	DEACTIVATE A UNIT
PRINT	PRINT STATISTICAL INFORMATION (IF IMPLEMENTED BY THE DIAGNOSTIC - SECTION 4.0)
DISPLAY	TYPE A LIST OF ALL DEVICE INFORMATION
FLAGS	TYPE THE STATE OF ALL FLAGS (SEE SECTION 2.3)
ZFLAGS	CLEAR ALL FLAGS (SEE SECTION 2.3)

A COMMAND CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. SO YOU MAY, FOR EXAMPLE, TYPE "STA" INSTEAD OF "START".

2.2 SWITCHES

THERE ARE SEVERAL SWITCHES WHICH ARE USED TO MODIFY SUPERVISOR OPERATION. THESE SWITCHES ARE APPENDED TO THE LEGAL COMMANDS. ALL OF THE LEGAL SWITCHES ARE TABULATED BELOW WITH A BRIEF DESCRIPTION OF EACH. IN THE DESCRIPTIONS BELOW, A DECIMAL NUMBER IS DESIGNATED BY "DDDDD".

SWITCH	EFFECT
/TESTS:LIST	EXECUTE ONLY THOSE TESTS SPECIFIED IN THE LIST. LIST IS A STRING OF TEST NUMBERS, FOR EXAMPLE - /TESTS:1:5:7-10. THIS LIST WILL CAUSE TESTS 1,5,7,8,9,10 TO BE RUN. ALL OTHER TESTS WILL NOT BE RUN.

/PASS:DDDDD	EXECUTE DDDDD PASSES (DDDDD = 1 TO 64000)
/FLAGS:FLGS	SET SPECIFIED FLAGS. FLAGS ARE DESCRIBED IN SECTION 2.3.
/EOP:DDDDD	REPORT END OF PASS MESSAGE AFTER EVERY DDDDD PASSES ONLY. (DDDDD = 1 TO 64000)
/UNITS:LIST	TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED IN THE LIST. LIST EXAMPLE - /UNITS:0:5:10-12 USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)

EXAMPLE OF SWITCH USAGE:

START/TESTS:1-5/PASS:1000/EOP:100

THE EFFECT OF THIS COMMAND WILL BE: 1) TESTS 1 THROUGH 5 WILL BE EXECUTED, 2) ALL UNITS WILL TESTED 1000 TIMES AND 3) THE END OF PASS MESSAGES WILL BE PRINTED AFTER EACH 100 PASSES ONLY. A SWITCH CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. YOU MAY, FOR EXAMPLE, TYPE "/TES:1-5" INSTEAD OF "/TESTS:1-5".

BELOW IS A TABLE THAT SPECIFIES WHICH SWITCHES CAN BE USED BY EACH COMMAND.

	TESTS	PASS	FLAGS	EOP	UNITS
START	X	X	X	X	X
RESTART	X	X	X	X	X
CONTINUE		X	X	X	
PROCEED			X		
DROP					X
ADD					X
PRINT					
DISPLAY					X
FLAGS					
ZFLAGS					
EXIT					

2.3 FLAGS

FLAGS ARE USED TO SET UP CERTAIN OPERATIONAL PARAMETERS SUCH AS LOOPING ON ERROR. ALL FLAGS ARE CLEARED AT STARTUP AND REMAIN CLEARED UNTIL EXPLICITLY SET USING THE FLAGS SWITCH. FLAGS ARE ALSO CLEARED AFTER A START COMMAND UNLESS SET USING THE FLAG SWITCH. THE ZFLAGS COMMAND MAY ALSO BE USED TO CLEAR ALL FLAGS. WITH THE EXCEPTION OF THE START AND ZFLAGS COMMANDS, NO COMMANDS AFFECT THE STATE OF THE FLAGS; THEY REMAIN SET OR CLEARED AS SPECIFIED BY THE LAST FLAG SWITCH.

FLAG	EFFECT
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBR*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)

IXR*	INHIBIT EXTENDED ERROR REPORTS (THOSE CALLED BY PRINTX MACRO'S)
PRI	DIRECT MESSAGES TO LINE PRINTER
PNT	PRINT TEST NUMBER AS TEST EXECUTES
BOE	"BELL" ON ERROR
UAM	UNATTENDED MODE (NO MANUAL INTERVENTION)
ISR	INHIBIT STATISTICAL REPORTS (DOES NOT APPLY TO DIAGNOSTICS WHICH DO NOT SUPPORT STATISTICAL REPORTING)
IDR	INHIBIT PROGRAM DROPPING OF UNITS
ADR	EXECUTE AUTODROP CODE
LOT	LOOP ON TEST
EVL	EXECUTE EVALUATION (ON DIAGNOSTICS WHICH HAVE EVALUATION SUPPORT)

*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

SEE THE XXDP+ USER'S MANUAL FOR MORE DETAILS ON FLAGS. YOU MAY SPECIFY MORE THAN ONE FLAG WITH THE FLAG SWITCH. FOR EXAMPLE, TO CAUSE THE PROGRAM TO LOOP ON ERROR, INHIBIT ERROR REPORTS AND TYPE A "BELL" ON ERROR, YOU MAY USE THE FOLLOWING STRING:

```
/FLAGS:LOE:IER:BOE
```

2.4 HARDWARE QUESTIONS

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING "CHANGE HW (L) ?" YOU MUST ANSWER "Y" AFTER A START COMMAND UNLESS THE HARDWARE INFORMATION HAS BEEN "PRELOADED" USING THE SETUP UTILITY (SEE CHAPTER 6 OF THE XXDP+ USER'S MANUAL). WHEN YOU ANSWER THIS QUESTION WITH A "Y", THE RUNTIME SERVICES WILL ASK FOR THE NUMBER OF UNITS (IN DECIMAL). YOU WILL THEN BE ASKED THE FOLLOWING QUESTIONS FOR EACH UNIT.

THE PROGRAM WILL ASK THE FOLLOWING QUESTIONS IN RESPONSE TO A START COMMAND (NON-SCRIPT):

CHANGE HW?

ANSWER NO TO USE THE PRE-BUILT ANSWERS FOR ALL HARDWARE QUESTIONS. THIS PROGRAM WILL BE RELEASED PRE-BUILT TO TEST ONE UNIT WITH THE DEFAULT ANSWERS SHOWN BELOW. THE PRE-BUILT ANSWERS MAY BE CHANGED AT ANY TIME WITH THE SETUP UTILITY. ANSWER 'YES' TO BE ASKED ALL THE HARDWARE QUESTIONS.

UNITS (D) ?

ANSWER WITH THE NUMBER OF UNITS TO BE TESTED (NO DEFAULT). THIS ANSWER WILL DETERMINE HOW MANY TIMES THE FOLLOWING QUESTIONS ARE ASKED. A UNIT IS A LOGICAL DISK DRIVE ON A UDA. 1 TO 64 UNITS MAY BE SPECIFIED.

UNIBUS ADDRESS OF UDA (O) 172150 ?

ANSWER WITH THE ADDRESS OF THE UDAIP REGISTER OF ONE UDA AS ADDRESSED BY THE PROCESSOR WITH MEMORY MANAGEMENT TURNED OFF (I.E., AN EVEN 16-BIT ADDRESS IN THE RANGE OF 160000 TO 177774).

VECTOR (O) 154 ?

ANSWER WITH THE INTERRUPT VECTOR ADDRESS OF THE UDA. A VECTOR ADDRESS IN THE RANGE OF 4 TO 774 MAY BE SPECIFIED.

BR LEVEL (D) 5 ?

ANSWER WITH THE INTERRUPT PRIORITY USED BY THE UDA. LEVELS 4 TO 7 ARE ACCEPTED.

UNIBUS BURST RATE (D) 0 ?

THE UDA ALLOWS THE ABILITY TO CONTROL THE MAXIMUM NUMBER OF WORDS TRANSFERRED ACROSS THE UNIBUS EACH TIME THE UDA BECOMES MASTER. ANSWER WITH THE VALUE YOUR OPERATING SYSTEM USES OR USE ZERO WHICH WILL TELL THE UDA TO SUPPLY A VALUE. A DECIMAL NUMBER IN THE RANGE OF 0 TO 63 MAY BE SPECIFIED. THE VALUE WILL BE PASSED DIRECTLY TO THE UDA DURING INITIALIZATION.

DRIVE NUMBER (D) 0 ?

ANSWER WITH THE LOGICAL DRIVE NUMBER ON THE FRONT OF THE DRIVE YOU WISH TO TEST. ON A MULTI-UNIT DRIVE, EACH SUB-UNIT NUMBER ON THE DRIVE MUST BE TESTED AS A SEPARATE UNIT TO COMPLETELY TEST THE DRIVE. A MAXIMUM OF EIGHT LOGICAL DRIVES MAY BE TESTED ON ONE UDA AT A TIME (UDA CONFIGURATION LIMIT).

EXERCISE ON CUSTOMER DATA AREA IN TEST 4 (L) N ?

ANSWER NO TO HAVE TEST 4 (DRIVE EXERCISER) RUN ON THE DIAGNOSTIC AREA OF THE DISK. ANSWER YES TO RUN ON THE CUSTOMER DATA AREA. A YES ANSWER WILL DESTROY ANY CUSTOMER DATA THAT MAY BE ON THE DISK. A WARNING MESSAGE WILL BE PRINTED IF THIS QUESTION IS ANSWERED YES (SEE SECTION 4.3).

2.5 SOFTWARE QUESTIONS

AFTER YOU HAVE ANSWERED THE HARDWARE QUESTIONS OR AFTER A RESTART OR CONTINUE COMMAND, THE RUNTIME SERVICES WILL ASK FOR SOFTWARE PARAMETERS. THESE PARAMETERS WILL GOVERN SOME DIAGNOSTIC SPECIFIC OPERATION MODES. YOU WILL BE PROMPTED BY "CHANGE SW (L) ?" IF YOU WISH TO CHANGE ANY PARAMETERS, ANSWER BY TYPING "Y". THE SOFTWARE QUESTIONS AND THE DEFAULT VALUES ARE DESCRIBED IN THE NEXT PARAGRAPH(S).

THE PROGRAM WILL ASK THE FOLLOWING QUESTIONS IN RESPONSE TO THE START,

RESTART, AND CONTINUE COMMANDS.

CHANGE SW ?

ANSWER NO TO BYPASS THE FOLLOWING QUESTIONS IN THIS SECTION. A YES ANSWER WILL CAUSE THE QUESTIONS TO BE ASKED AND ALLOW THE DEFAULT ANSWERS TO BE CHANGED.

ENTER MANUAL INTERVENTION MODE FOR SPECIAL DIAGNOSIS (L) N ?

TESTS 2 AND 4 HAVE MANUAL INTERVENTION MODES WHICH ALLOW ADDITIONAL PARAMETERS TO BE INPUT TO ALTER THE NORMAL TESTING OF A DISK DRIVE. THIS QUESTION SHOULD NORMALLY BE ANSWERED NO WHEN THIS DIAGNOSTIC IS FIRST RUN. THEN, DEPENDING ON THE ERRORS DETECTED, IT MAY BE DESIRABLE TO CHANGE THIS ANSWER TO YES AND ALTER THE TESTING TO FURTHER ISOLATE THE PROBLEM. IF THIS QUESTION IS ANSWERED YES, AND THE UAM (UNATTENDED MODE OPERATION) FLAG IS SET, TESTS 2 AND 4 WILL PRINT A WARNING MESSAGE THAT THE MODE CANNOT BE ENTERED AND WILL PROCEED AS IF ANSWERED NO. SEE THE DESCRIPTION OF THE INDIVIDUAL TESTS FOR MORE INFORMATION.

REMAINING SOFTWARE QUESTIONS APPLY TO TEST 4 ONLY

THIS INFORMATIONAL MESSAGE IS PRINTED TO DESCRIBE THE USE OF THE REMAINING QUESTIONS. IF TEST 4 IS NOT BEING RUN, A ''CONTROL Z'' CAN BE TYPED TO BYPASS THEM.

ERROR LIMIT (D) 32 ?

ENTER THE NUMBER OF HARD ERRORS ALLOWED BEFORE A DRIVE IS DROPPED FROM EXERCISE BY TEST #4. A NUMBER IN THE RANGE OF 1 TO 65535 WILL BE ACCEPTED.

READ TRANSFER LIMIT IN MEGABYTES - C FOR NO LIMIT (D) 0 ?

WHEN THE SPECIFIED NUMBER OF BYTES HAVE BEEN READ FROM A DRIVE BY TEST #4, THE DRIVE WILL BE DROPPED FROM TESTING. WHEN ALL DRIVES ARE DROPPED, AN END OF PASS WILL BE INDICATED AND THE SELECTED TESTS WILL BE RUN AGAIN. THIS IS THE METHOD USED TO DETERMINE HOW LONG TEST #4 IS TO RUN. ANSWER WITH A ZERO TO PREVENT TEST FROM ENDING. THE ONLY OTHER WAY TEST #4 CAN END IS TO HAVE ALL DRIVES DROPPED BECAUSE THE ERROR LIMIT ON EACH IS EXCEEDED. OF COURSE, THE OPERATOR CAN ALWAYS STOP TEST #4 BY TYPING A CONTROL-C.

SUPPRESS PRINTING SOFT ERRORS (L) Y?

WHEN TEST #4 NEEDS TO PERFORM RETRIES, SOFT ERROR REPORTS WILL BE PRINTED TO GIVE AS MUCH INFORMATION AS POSSIBLE. THESE ACTIONS ARE CONSIDERED NORMAL OPERATION AND ARE NOT ERROR CONDITIONS UNTIL THE RETRIES FAIL. WHEN THE TEST IS BEING RUN ONLY TO SEE HOW RELIABLE THE DRIVE PERFORMS, THIS QUESTION SHOULD BE ANSWERED YES SO THEY ARE NOT CONFUSED WITH HARD ERRORS. THE NUMBER OF THESE SOFT ERRORS IS ALWAYS REPORTED IN THE STATISTICAL REPORT. ANSWER NO TO SEE ALL THE SOFT ERROR REPORTS.

DO INITIAL WRITE ON START (L) Y ?

IF TEST #4 IS TO DO DATA COMPARES, THE DRIVE WILL NEED TO BE WRITTEN WITH DATA PATTERNS READABLE BY THE PROGRAM.

IF THE DIAGNOSTIC AREA IS SELECTED FOR TESTING, THE INITIAL WRITE IS ALWAYS PERFORMED (REGARDLESS OF HOW THIS QUESTION IS ANSWERED).

IF THE CUSTOMER DATA AREA IS SELECTED FOR TESTING, THE INITIAL WRITE WILL BE PERFORMED WHEN ALL OF THE FOLLOWING ARE TRUE:

1. THIS QUESTION IS ANSWERED YES.

2. THIS IS THE FIRST TIME TEST #4 IS BEING RUN AFTER A START COMMAND.

3. THE DISK IS WRITE ENABLED.

ANSWERING THIS QUESTION NO WHEN TESTING ON THE CUSTOMER DATA AREA WILL NORMALLY RESULT IN DATA COMPARISON ERRORS IF THE DISK WAS NOT PREVIOUSLY WRITTEN BY THIS DIAGNOSTIC OR THE FORMATTER.

NOTE THAT WRITE CHECKS ARE NOT PERFORMED DURING THE INITIAL WRITE.

ENABLE ERROR LOG (L) N ?

A YES ANSWER WILL CAUSE ERROR MESSAGES IN TEST #4 TO BE STORED IN A LOG BUFFER. ONCE THE LOG BUFFER IS FULL, ADDITIONAL ERROR INFORMATION IS LOST. THE CONTENTS OF THE LOG BUFFER WILL BE PRINTED WHEN TEST #4 IS STOPPED AND A STATISTICAL REPORT REQUESTED. THIS LOG FEATURE IS INTENDED TO ALLOW THE DIGITAL DIAGNOSIS CENTER (DDC) TO START TEST #4 THEN HANG UP FROM THE SYSTEM AND LET IT RUN FOR SOME PERIOD OF TIME. DDC CAN CALL THE SYSTEM BACK LATER, TYPE CONTROL-C, THEN PRINT AND SEE THE ERRORS THAT HAVE OCCURRED (UP TO THE LIMIT OF THE LOG BUFFER). A MESSAGE WILL BE PRINTED TO INDICATE NO ERRORS HAVE OCCURRED IF THE LOG BUFFER IS EMPTY. TEST #4 WILL NOT BE ALLOWED TO END WHILE THE ERROR LOG IS ENABLED EVEN THOUGH TESTING MAY ACTUALLY STOP. THE LOG BUFFER WILL HOLD A MINIMUM OF 30 ERROR MESSAGES.

2.6 EXTENDED P-TABLE DIALOGUE

THE FOLLOWING WARNING WILL BE PRINTED IN RESPONSE TO A START COMMAND IF ANY DRIVE IS SELECTED FOR TESTING ON THE CUSTOMER DATA AREA. THE WARNING WILL APPEAR IMMEDIATELY AFTER THE SOFTWARE QUESTIONING IS COMPLETED.

CUSTOMER DATA WILL BE DESTROYED ON:

UNIT	UDA AT	DRIVE
XX	XXXXXX	XXX

UNLESS THE DIAGNOSTIC IS BEING RUN IN UNATTENDED MODE (E.G., START/FLAG:UAM TO SUPERVISOR PROMPT), A CONFIRMATION WILL ALSO BE REQUIRED AS FOLLOWS:

ARE YOU SURE CUSTOMER DATA CAN BE DESTROYED (L) ?

IF THE ABOVE QUESTION IS ANSWERED NO, THE ENTIRE DIAGNOSTIC WILL STOP AND THE SUPERVISOR PROMPT WILL BE DISPLAYED. NO DEFAULT ANSWER IS PROVIDED FOR THIS QUESTION.

2.7 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

. BOOT XXDP+

2. GIVE THE DATE AND ANSWER THE LSI AND 50HZ (IF THERE IS A CLOCK) QUESTIONS

3. TYPE 'R NAME', WHERE NAME IS THE NAME OF THE BIN OR BIC FILE FOR THIS PROGRAM

4. TYPE ''START''

5. ANSWER THE ''CHANGE HW'' QUESTION WITH ''Y''

6. ANSWER ALL THE HARDWARE QUESTIONS

7. ANSWER THE ''CHANGE SW'' QUESTION WITH ''N''

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING ONLY THE DEFAULTS FOR FLAGS AND SOFTWARE PARAMETERS. THESE DEFAULTS ARE DESCRIBED IN SECTIONS 2.3 AND 2.5.

3.0 ERROR INFORMATION

3.1 TYPES OF ERROR MESSAGES

THERE ARE THREE LEVELS OF ERROR MESSAGES THAT MAY BE ISSUED BY A DIAGNOSTIC: GENERAL, BASIC AND EXTENDED. GENERAL ERROR MESSAGES ARE ALWAYS PRINTED UNLESS THE ''IER'' FLAG IS SET (SECTION 2.3). THE GENERAL ERROR MESSAGE IS OF THE FORM:

NAME TYPE NUMBER ON UNIT NUMBER TST NUMBER PC:XXXXXX
ERROR MESSAGE

,WHERE; NAME = DIAGNOSTIC NAME
TYPE = ERROR TYPE (SYS FATAL, DEV FATAL, HARD OR SOFT)
NUMBER = ERROR NUMBER
UNIT NUMBER = 0 - N (N IS LAST UNIT IN PTABLE)
TST NUMBER = TEST AND SUBTEST WHERE ERROR OCCURRED
PC:XXXXXX = ADDRESS OF ERROR MESSAGE CALL

BASIC ERROR MESSAGES ARE MESSAGES THAT CONTAIN SOME ADDITIONAL INFORMATION ABOUT THE ERROR. THESE ARE ALWAYS PRINTED UNLESS THE "IER" OR "IBR" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL MESSAGE.

EXTENDED ERROR MESSAGES CONTAIN SUPPLEMENTARY ERROR INFORMATION SUCH AS REGISTER CONTENTS OR GOOD/BAD DATA. THESE ARE ALWAYS PRINTED UNLESS THE "IER", "IBR" OR "IXR" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL ERROR MESSAGE AND ANY ASSOCIATED BASIC ERROR MESSAGES.

4.0 PERFORMANCE AND PROGRESS REPORTS

A STATISTICAL REPORT WILL AUTOMATICALLY BE PRINTED PERIODICALLY (APPROXIMATELY EVERY SEVENTEEN MINUTES) AND AT THE END OF TEST #4. IT CAN BE SUPPRESSED BY SETTING THE INHIBIT STATISTICAL REPORT FLAG (E.G. START/FLAGS:ISR). THIS IS THE SAME REPORT THAT CAN BE PRINTED ON DEMAND WITH THE PRINT COMMAND.

DURING TEST 1, 2, AND 3, THE REPORT WILL LOOK LIKE THE FOLLOWING EXAMPLE:

TEST 1 IN PROGRESS RUN TIME 2:24:10

DURING TEST #4, THE REPORT WILL CONTAIN STATISTICS ON EACH DRIVE FOR THE CURRENT PASS OF THE TEST; FOR EXAMPLE:

TEST 4 IN PROGRESS RUN TIME 2:24:10

UNIT	DRIVE	SERIAL-NUMBER	SEEKS	MBYTES READ	MBYTES WRITTEN	HARD ERRORS	SOFT ERRORS	ECC
0	0	1002	12000	36	22	0	0	1
1	4	67342102112	14000	42	29	0	2	0

5.0 TEST SUMMARIES

TEST # 1 - UNIBUSS ADDRESSING TEST

THE PURPOSE OF TEST #1 IS TO COMPLETE THE TESTING OF THE UNIBUS INTERFACE IN THE UDA. THE UDA RESIDENT DIAGNOSTIC IS NOT ABLE TO COMPLETELY TEST THE UNIBUS INTERFACE BECAUSE COMMUNICATION WITH THE PDP-11 PROCESSOR IS NECESSARY. SPECIFICALLY, THIS TEST WILL:

1. CHECK THAT EVERY ADDRESS LINE ON THE UNIBUS CAN BE DRIVEN TO BOTH ONE AND ZERO STATES.

2. CHECK THAT THE UDA CAN INTERRUPT THE PDP-11 PROCESSOR AT THE PROPER PRIORITY LEVEL AND VECTOR.

3. EXERCISE THE UNIBUS INTERFACE BY TRANSFERRING BLOCKS OF DATA TO AND FROM UNIBUS MEMORY.

THIS TEST ASSUMES THAT THE FOLLOWING ARE BEING TESTED BY THE UDA RESIDENT DIAGNOSTIC:

1. ALL DATA BITS CAN BE WRITTEN AND READ CORRECTLY.

2. NPR CYCLES CAN BE EXECUTED CORRECTLY.

ONE AT A TIME, EACH UDA SELECTED FOR TESTING WILL BE INITIALIZED AND BROUGHT ON-LINE BY FOLLOWING THE INITIALIZATION PROTOCOL. SEVERAL SIZES OF THE HOST COMMUNICATIONS AREA WILL BE SUPPLIED TO ALLOW THE UDA RESIDENT DIAGNOSTIC TO DO THE MOST UNIBUS ADDRESS TESTING POSSIBLE. INTERRUPTS WILL BE DISABLED. ANY UDA RESIDENT DIAGNOSTIC ERRORS WILL BE REPORTED.

THE UDA WILL THEN BE INITIALIZED AGAIN, THIS TIME WITH INTERRUPTS ENABLED. THE VECTOR ADDRESS AND PRIORITY LEVEL WILL BE DETERMINED SOLELY FROM THE ANSWERS TO THE HARDWARE QUESTIONS. IF THE HARDWARE VECTORS TO THE WRONG ADDRESS, IT IS IMPOSSIBLE TO DETERMINE THE RESULT. A DESCRIPTIVE ERROR MESSAGE OF THE PROBLEM WILL NOT OCCUR (THE PROGRAM OR PROCESSOR MAY HANG OR AN UNRELATED MESSAGE MAY OCCUR). THEREFORE, THE MESSAGE "TESTING INTERRUPT ABILITY OF UDA AT ADR XXXXXX VEC XXX..." WILL BE PRINTED JUST BEFORE THE UDA IS REQUESTED TO CAUSE AN INTERRUPT AND THE WORD "COMPLETED" WILL BE PRINTED (ON THE SAME LINE) WHEN THE INTERRUPT TEST IS COMPLETED. IF THE WORD "COMPLETED" DOES NOT FOLLOW THE FIRST MESSAGE, IT SHOULD BE APPARENT THAT THE INTERRUPT CAUSED THE DIAGNOSTIC OR PROCESSOR TO GO ASTRAY. THE PRIORITY LEVEL OF THE INTERRUPT REQUEST WILL ALSO BE VERIFIED.

A "DIAGNOSTIC MACHINE" PROGRAM WILL THEN BE DOWNLINE LOADED IN THE UDA FROM THE MEMORY SPACE INCLUDED IN THE HOST COMMUNICATIONS AREA WHEN THE UDA WAS FIRST INITIALIZED. THE UDA RESIDENT DIAGNOSTIC HAS ALREADY VERIFIED THAT IT CAN ACCESS THESE MEMORY ADDRESSES, SO THE DOWNLINE LOAD COMMAND SHOULD PERFORM PROPERLY. THE "DIAGNOSTIC MACHINE" PROGRAM IS THEN STARTED.

THE "DIAGNOSTIC MACHINE" PROGRAM WILL ASK THE PDP-11 PROGRAM TO FILL FREE MEMORY (THAT MEMORY AVAILABLE TO THE PDP-11 PROGRAM THAT IS NOT BEING USED BY THE PROGRAM OR THE PDP-11 SUPERVISOR) WITH AN ADDRESSING PATTERN AND REPORT THE LOCATION AND SIZE OF THE FREE MEMORY. EVERY LOCATION OF FREE MEMORY WILL BE READ AND THE DATA CHECKED. THEN, ONE BY ONE, EACH ADDRESS LINE WILL BE TESTED AS FOLLOWS:

1. DETERMINE A TEST ADDRESS BY TAKING THE FIRST ADDRESS OF FREE MEMORY AND COMPLIMENTING THE ADDRESS BIT TO BE TESTED.

2. READ FROM THE TEST ADDRESS.

3. IF A NON-EXISTANT MEMORY ERROR OCCURRS, THE TEST IS COMPLETE.

4. WRITE ALL ONES TO THE FIRST ADDRESS OF FREE MEMORY THEN READ FROM THE TEST ADDRESS. IF DATA READ IS NOT ALL ONES, THEN TEST IS COMPLETE.

5. WRITE ZEROS TO THE FIRST ADDRESS OF FREE MEMORY THEN READ FROM THE TEST ADDRESS. IF DATA READ IS NOT ZEROS, THEN TEST IS COMPLETE.

6. REPORT UNIBUS ADDRESSING ERROR.

WHEN ALL ADDRESS BITS HAVE BEEN TESTED, THEN BLOCK TRANSFERS TO AND

FROM MEMORY WILL BE TESTED WITH DIFFERENT DATA PATTERNS. THIS DATA WILL BE TRANSFERRED AT THE RATE DISK DATA IS TRANSFERRED TO AND FROM MEMORY DURING NORMAL UDA OPERATION.

THE NEXT UDA SELECTED FOR TESTING WILL THEN BE TESTED IN THE SAME MANNER. WHEN ALL UDAS HAVE BEEN TESTED, TEST #1 WILL END.

TEST #2 - DISK RESIDENT DIAGNOSTIC TEST

THE PURPOSE OF TEST #2 IS TO EXECUTE THE DIAGNOSTICS THAT RUN IN EACH DISK DRIVE. THESE DIAGNOSTIC PROGRAMS MAY BE RESIDENT IN THE DISK DRIVE OR REQUIRE DOWNLINE LOADING FROM THE XXDP+ LOAD DEVICE. THESE DIAGNOSTIC PROGRAMS THAT RUN IN THE DISK DRIVES ARE NOT PART OF THIS DIAGNOSTIC PRODUCT, BUT ARE PRODUCED BY THE DISK DEVELOPMENT GROUP. THIS UDA DIAGNOSTIC PROGRAM ONLY KNOWS THE PROCEDURE TO EXECUTE THE DISK RESIDENT DIAGNOSTICS AND HOW TO DETERMINE WHETHER A TEST PASSED OR FAILED.

ONE AT A TIME, EACH UDA SELECTED FOR TESTING WILL BE INITIALIZED AND A ''DIAGNOSTIC MACHINE'' PROGRAM DOWNLINE LOADED. THE ''DIAGNOSTIC MACHINE'' PROGRAM WILL ASK WHAT DRIVES ARE TO BE TESTED, THEN WILL ISSUE SEVERAL ECHO FRAMES TO THE DISK DRIVE AND CHECK FOR THE CORRECT RESPONSE FROM THE DRIVE. THIS SHOULD SERVE AS A GOOD INDICATOR THAT THE UDA AND DISK DRIVE CAN COMMUNICATE.

A DIAGNOSE COMMAND WILL THEN BE ISSUED TO THE DRIVE TO REQUEST THE DRIVE RUN ALL OF ITS DIAGNOSTICS. IF THE DISK DRIVE REQUESTS A DOWNLINE LOAD OF A DRIVE DIAGNOSTIC, THE DIAGNOSTIC PROGRAM WILL BE READ FROM THE XXDP+ LOAD DEVICE, DOWNLINE LOAD THE FILE INTO THE DISK DRIVE AND START ITS EXECUTION. THERE IS NO LIMIT TO THE NUMBER OF DOWNLINE LOADS THAT CAN BE REQUESTED BY A DRIVE.

IF THE ''MANUAL INTERVENTION MODE'' SOFTWARE QUESTION WAS ANSWERED NO (DEFAULT) TESTING WILL PROCEED TO THE NEXT DRIVE. WHEN ALL DRIVES ON THE UDA HAVE BEEN TESTED, THE NEXT UDA SELECTED FOR TESTING WILL THEN BE TESTED IN THE SAME MANNER. WHEN ALL UDA'S HAVE BEEN TESTED, TEST #2 WILL END.

IF THE ''MANUAL INTERVENTION MODE'' SOFTWARE QUESTION WAS ANSWERED YES, AN INTERACTIVE MODE WILL BE ENTERED TO ALLOW THE OPERATOR TO PERFORM DIAGNOSTIC ACTIVITIES ON THE DISK DRIVE AS DESIRED. THE SERVICE MANUAL FOR THE DISK DRIVE MUST BE USED TO DETERMINE WHAT DIAGNOSTIC CAPABILITIES ARE AVAILABLE.

FIRST, A BRIEF DESCRIPTION OF AVAILABLE COMMANDS WILL BE PRINTED AS FOLLOWS:

```
TEST #2 MANUAL INTERVENTION ON UNIT XX UDA AT XXXXXX DRIVE XXX
TO WRITE AND READ MEMORY:
  W DATA REGION OFFSET
  R REGION OFFSET
TO RUN A DIAGNOSTIC:
  D REGION
TO EXIT QUESTIONING:
```

E
DATA, REGION AND OFFSET ARE HEX VALUES.
 ?

COMMANDS MAY BE TYPED AFTER THE QUESTION MARK PROMPT. EACH COMMAND WILL BE PROCESSED AS ENTERED AND RESULTS DISPLAYED IMMEDIATELY. THE EXIT COMMAND WILL ALLOW THE DIAGNOSTIC TO PROCEED.

READ AND WRITE COMMANDS WILL REMEMBER THE REGION AND OFFSET VALUES. SUCCESSIVE READ AND SUCCESSIVE WRITE COMMANDS WILL AUTOMATICALLY INCREMENT TO THE NEXT OFFSET IF THE REGION AND OFFSET VALUES ARE NOT TYPED. IF A REGION IS TYPED BUT NOT AN OFFSET, OFFSET ZERO WILL BE USED.

ONE TO FOUR BYTES OF DATA MAY BE ENTERED BY A SINGLE WRITE COMMAND, DEPENDING ON THE NUMBER OF DIGITS TYPED IN THE HEX VALUE. A READ COMMAND WILL ALWAYS RETURN FOUR BYTES OF DATA. EXAMPLES:

```
1.      W FF FFFC 4
2.      W 010203
3.      R FFFC 0004
          FFFC 0004/ FF 01 02 03
4.      W F0F1F2F3 FFFC
5.      R
          FFFC 0000/ F0 F1 F2 F3
6.      R
          FFFC 0004/ FF 01 02 03
```

COMMAND 1 WRITES ONE BYTE (FF) INTO REGION FFFC, OFFSET 4. COMMAND 2 WRITES THREE BYTES (01, 02 AND 03) INTO THE SAME REGION WITH OFFSETS 5, 6 AND 7. COMMAND 3 READS FOUR BYTES STARTING AT REGION FFFC OFFSET 4. COMMAND 4 WRITES FOUR BYTES AT REGION FFFC OFFSET 0. COMMANDS 5 AND 6 READ THE EIGHT BYTES.

THE DIAGNOSE COMMAND WILL REMEMBER THE REGION FROM PREVIOUS DIAGNOSE COMMANDS ONLY, BECAUSE THE REGION CONTAINING THE DIAGNOSTIC IS GENERALLY NOT THE SAME REGION USED TO WRITE PARAMETERS OR READ RESULTS. IF THE DIAGNOSTIC RETURNS ANY DATA, THE DATA WILL BE PRINTED IMMEDIATELY.

TEST #3 - DISK FUNCTION TEST

THE PURPOSE OF TEST #3 IS TO FUNCTIONALLY TEST THE DISK DRIVE. ON A DRIVE THAT IS WELL DIAGNOSED BY ITS DISK RESIDENT DIAGNOSTICS (EXECUTED BY TEST #2) THESE FUNCTIONAL TESTS WILL HAVE LITTLE VALUE. ON A DRIVE THAT HAS NO OR MINIMAL RESIDENT DIAGNOSTICS, THESE FUNCTIONAL TESTS WILL HAVE MORE VALUE.

TEST #3 WILL START BY INITIALIZING EACH UDA SELECTED FOR TESTING AND THEN DOWNLINE LOAD A "DIAGNOSTIC MACHINE" PROGRAM INTO EACH UDA. ONCE ALL UDAS HAVE BEEN STARTED, THE PDP-11 PROGRAM WILL RESPOND TO REQUESTS FROM ALL UDAS. WHEN ALL THE UDAS HAVE INDICATED THE END OF TESTING, TEST #3 WILL END.

THE "DIAGNOSTIC MACHINE" PROGRAM WILL PERFORM THE FOLLOWING FUNCTIONS

ON EACH DRIVE:

1. ISSUE A DRIVE CLEAR COMMAND.

2. ISSUE INITIATE RECALIBRATE COMMAND.

3. ISSUE A CHANGE MODE COMMAND TO ENABLE DIAGNOSTIC CYLINDER ACCESS AND SET THE DRIVE TO 512 BYTE SECTOR SIZE.

4. ISSUE INITIATE SEEK COMMAND TO LAST DIAGNOSTIC CYLINDER.

5. READ ALL FACTORY FORMATTED SECTOR HEADERS. IF NO HEADERS ON A TRACK CAN BE READ, REPORT THE ERROR, OTHERWISE CONTINUE.

6. STARTING WITH CYLINDER 0, GROUP 0 AND INCREMENTING THROUGH EVERY GROUP ON THE DISK, SEEK TO THE SELECTED GROUP, READ A HEADER ON TRACK 0 AND THEN SEEK TO THE FACTORY FORMATTED DIAGNOSTIC CYLINDER. READ FROM THE DIAGNOSTIC CYLINDER TO VERIFY DISK POSITIONED CORRECTLY.

7. ISSUE A CHANGE MODE COMMAND TO ENABLE FORMATTING OPERATIONS.

8. FORMAT ALL WRITABLE DBNS IN 512 BYTE FORMAT.

9. WRITE AND READ SEVERAL DATA PATTERNS TO EACH WRITABLE DBN. REPORT AN ERROR IF ALL DBNS ON ONE TRACK HAVE AN ERROR.

TEST #4 - DISK EXERCISER

THE PURPOSE OF TEST #4 IS TO EXERCISE THE DISK DRIVES IN A MANNER SIMILAR TO NORMAL USAGE UNDER STANDARD OPERATING SYSTEMS. EXECUTION OF THIS TEST SHOULD GIVE AN INDICATION OF THE PERFORMANCE OF THE DISK DRIVE. THIS TEST MAY BE RUN FOR LONG OR SHORT PERIODS OF TIME, DEPENDING ON HOW THE SOFTWARE QUESTIONS ARE ANSWERED.

THESE ARE TWO MODES OF OPERATION FOR TEST #4:

1. DEFAULT OPERATION ON THE ENTIRE AREA SELECTED (CUSTOMER OR DIAGNOSTIC) WITH ALL PARAMETERS SELECTED FOR RANDOM OPERATION AS SHOWN BY DEFAULT ANSWERS BELOW.

2. MANUAL INTERVENTION MODE WHERE A NUMBER OF QUESTIONS ARE ASKED AND OPERATION IS CONTROLLED BY THEIR ANSWERS.

WHICH MODE IS ENTIRELY DETERMINED BY THE ANSWER TO THE FIRST SOFTWARE QUESTION ASKING, 'ENTER MANUAL INTERVENTION MODE FOR SPECIAL DIAGNOSIS?'' THIS QUESTION WOULD NORMALLY HAVE BEEN ANSWERED NO (DEFAULT) AND TESTING WILL BEGIN IMMEDIATELY. IF ANSWERED YES, THE FOLLOWING SERIES OF QUESTIONS WILL BE ASKED FOR EACH UNIT SELECTED FOR TESTING:

THE FOLLOWING QUESTIONS REFER TO UNIT XX UDA AT XXXXXX DRIVE XXX

THIS MESSAGE WILL IDENTIFY TO WHICH DRIVE THE QUESTIONS ARE BEING ASKED. THE ENTIRE SERIES OF QUESTIONS WILL BE ASKED FOR EACH DRIVE, THERE IS NO SHORT WAY TO ANSWER LIKE IN THE

HARDWARE QUESTIONS.

NUMBER OF BAD BLOCKS (D) 0 ?

AN ANSWER IN THE RANGE OF 1 TO 16 WILL ALLOW THAT MANY BAD BLOCK NUMBERS TO BE ENTERED. THE PROGRAM WILL ALLOW WRITES AND READS TO THESE BLOCKS BUT NO ERROR MESSAGES WILL BE PRINTED FOR THESE BLOCKS. ERRORS ENCOUNTERED ON THESE BLOCKS WILL NOT APPEAR IN THE STATISTICS. ANSWER ZERO TO BYPASS ENTERING BAD BLOCKS.

BAD BLOCK (A) ?

THIS QUESTION WILL BE ASKED THE NUMBER OF TIMES REQUESTED BY THE PREVIOUS ANSWER. ANY DECIMAL NUMBER THAT CAN BE CONVERTED INTO A 28-BIT BINARY VALUE WILL BE ACCEPTED. NO OTHER ERROR CHECKING WILL BE MADE AT THIS TIME TO DETERMINE IF THE BLOCK NUMBER ACTUALLY EXISTS ON THE DISK.

READ ONLY (L) N ?

ANSWER YES TO DICTATE READ ONLY AND PREVENT TEST #4 FROM PERFORMING ANY WRITES TO THE DISK. NOTE THAT TEST #3 WILL STILL WRITE TO THE DIAGNOSTIC CYLINDERS.

WRITE ONLY (L) N ?

THIS QUESTION WILL ONLY BE ASKED IF THE PREVIOUS QUESTION WAS ANSWERED NO. ANSWER YES TO DICTATE WRITE ONLY.

CHECK ALL WRITES BY READING (L) N ?

ANSWER YES TO CAUSE ALL WRITES TO BE CHECKED BY READING THE DATA IMMEDIATELY AFTER THE WRITE OPERATION.

RANDOMLY CHECK WRITES BY READING (L) Y ?

THIS QUESTION WILL ONLY BE ASKED IF THE PREVIOUS QUESTION WAS ANSWERED NO. ANSWER YES FOR THE WRITE CHECK TO BE PERFORMED RANDOMLY. ANSWER NO IF WRITE CHECKS ARE NOT DESIRED.

DATA PATTERN - 0 FOR RANDOM SELECTION (D) 0 ?

THERE ARE 16 DATA PATTERNS AVAILABLE, SELECTED AS 1 TO 16. PATTERN NUMBER 0 WILL CAUSE PATTERNS 1 TO 15 TO BE RANDOMLY SELECTED FOR EACH WRITE. IF PATTERN NUMBER 16 IS SELECTED, THE FOLLOWING SET OF QUESTIONS WILL BE ASKED FOR A PATTERN TO BE INPUT.

ENABLE ECC DATA CORRECTION (L) Y ?

A YES ANSWER WILL ENABLE THE USE OF ECC TO CORRECT DATA ERRORS. NO ERROR MESSAGE WILL BE PRINTED WHEN THE ECC PROPERLY CORRECTS THE DATA ON THE FIRST READ OF ANY DISK BLOCK. READ RETRIES WILL END WHEN A RE-READ PRODUCES AN ECC CORRECTABLE ERROR. THE USE OF ECC CORRECTION WILL APPEAR ONLY IN THE STATISTICAL REPORT FOR THE DRIVE.

A NO ANSWER WILL PREVENT THE USE OF ECC. ALL ECC ERRORS WILL CAUSE AN ERROR MESSAGE TO BE PRINTED AND RETRIES TO BE ATTEMPTED.

COMPARE ALL DATA READ (L) N ?

ANSWER YES TO CAUSE A DATA COMPARE AFTER EVERY READ.

RANDOMLY COMPARE DATA READ (L) Y ?

THIS QUESTION WILL ONLY BE ASKED IF THE PREVIOUS QUESTION WAS ANSWERED NO. ANSWER YES FOR THE DATA COMPARE TO BE PERFORMED ON RANDOM RECORDS. ANSWER NO IF DATA COMPARES ARE NOT DESIRED.

ENABLE RETRIES (L) Y

A YES ANSWER WILL ENABLE RETRIES TO BE PERFORMED ON DISK ERRORS.

RANDOM SEEK MODE (L) Y ?

ANSWER YES TO CAUSE BLOCK NUMBERS TO BE CHOSEN RANDOMLY. ANSWER NO TO CAUSE BLOCK NUMBERS TO BE SELECTED SEQUENTIALLY UP AND DOWN THE DISK SURFACE.

DO YOU WISH TO:
   0 - TEST ENTIRE AREA SELECTED
   1 - SPECIFY BEGIN/END SETS TO TEST
   2 - SPECIFY TRACKS AND CYLINDERS TO TEST
   3 - SPECIFY GROUPS AND CYLINDERS TO TEST
   4 - SPECIFY CYLINDERS TO TEST
(D) 0 ?

THIS QUESTION SPECIFIES THE OPTIONS AVAILABLE TO LIMIT TESTING TO A PORTION OF THE SELECTED AREA (CUSTOMER OR DIAGNOSTIC) OF THE DISK. A ZERO ANSWER IS THE DEFAULT WHICH SPECIFIES TO USE THE ENTIRE AREA FOR THE TEST. OTHER ANSWERS WILL CAUSE ADDITIONAL QUESTIONS TO BE ASKED.

NUMBER OF BEGIN/END SETS (D) 1 ?
BEGIN BLOCK (A) 0 ?
END BLOCK (A) 0 ?

THESE QUESTIONS ARE ASKED IF BEGIN/END SETS WERE SELECTED TO LIMIT THE TESTING AREA (ANSWER 1). ONE TO FOUR SETS MAY BE SPECIFIED. THE BEGIN BLOCK AND END BLOCK QUESTIONS ARE ASKED AS MANY TIMES AS NEEDED.

NUMBER OF TRACKS TO TEST (D) 1 ?
TRACK (D) 0 ?

NUMBER OF GROUPS TO TEST (D) 1 ?
GROUP (D) 0 ?

ONE OF THESE SETS OF QUESTIONS IS ASKED IF EITHER TRACKS AND

CYLINDERS OR GROUPS AND CYLINDERS WAS SPECIFIED TO LIMIT THE
TESTING AREA (ANSWERS 2 OR 3). UP TO SEVEN TRACKS OR GROUPS
MAY BE SPECIFIED ON WHICH TESTING WILL BE LIMITED.

DO YOU WISH TO LIMIT THE CYLINDERS TESTED (L) N ?

THIS QUESTION IS ASKED ONLY AFTER THE TRACKS OR GROUPS HAVE
BEEN SPECIFIED ABOVE. IF TESTING IS TO BE FURTHER LIMITED TO
A SET OF CYLINDERS, ANSWER YES AND THE FOLLOWING TWO QUES-
TIONS WILL BE ASKED:

STARTING CYLINDER (A) 0 ?
ENDING CYLINDER (A) 0 ?

THESE QUESTIONS ARE ASKED IF THE QUESTION IMMEDIATELY ABOVE
WAS ANSWERED YES OR IF CYLINDERS WERE SELECTED TO LIMIT THE
TESTING AREA (ANSWER 4). ONE SET OF CYLINDER NUMBERS MAY BE
SPECIFIED TO LIMIT THE TESTING AREA.

AFTER THE ABOVE QUESTIONS HAVE BEEN ASKED FOR ALL DRIVES SELECTED FOR
TESTING, THE FOLLOWING QUESTIONS WILL BE ASKED IF DATA PATTERN 16 WAS
SELECTED FOR ANY DRIVE:

NUMBER OF WORDS IN DATA PATTERN 16 (D) 1 ?
DATA WORD (O) 0 ?

DATA PATTERN 16 CAN BE INPUT BY THESE QUESTIONS. A DATA
PATTERN CONSISTS OF A BUFFER OF ONE TO 16 WORDS WHICH IS
REPEATED THROUGHOUT THE DATA PORTION OF THE DISK BLOCK. ENTER
THE CONTENTS OF THE DATA PATTERN BUFFER. THE DATA WORD
QUESTION WILL BE REPEATED AS NEEDED.

TEST #4 WILL START BY INITIALIZING EACH UDA SELECTED FOR TESTING AND
THEN DOWNLINE LOADING A "DIAGNOSTIC MACHINE" PROGRAM INTO EACH UDA.
THE "DIAGNOSTIC MACHINE" PROGRAM WILL ASK WHAT DRIVES ARE TO BE TESTED
AND THEN WILL ASK FOR THE PARAMETERS FOR EACH DRIVE (THE ANSWERS TO
THE MANUAL INTERVENTION QUESTIONS OR THEIR DEFAULTS). ONCE ALL UDAS
HAVE BEEN STARTED, THE PDP-11 PROGRAM WILL RESPOND TO REQUESTS FROM
ALL UDAS.

THE DISK WILL THEN BE EXERCISED ACCORDING TO THE PARAMETERS. THE
EXERCISE CONSISTS OF SELECTING A DISK SECTOR, SEEKING TO THE PROPER
CYLINDER, THEN READING OR WRITING THE SECTOR. THE PARAMETERS WILL
CONTROL HOW THE DISK SECTOR IS SELECTED, WHETHER THE SECTOR IS WRITTEN
OR READ AND WHETHER A WRITE IS FOLLOWED BY A READ (WRITE CHECK).

THE "DIAGNOSTIC MACHINE" PROGRAM WILL PERIODICALLY SEND STATISTICS TO
THE PDP-11 PROGRAM. THESE STATISTICS WILL INCLUDE COUNTS OF READS,
WRITES, SEEKS AND ERRORS ON A PER DRIVE BASIS. THE PDP-11 PROGRAM WILL
ACCUMULATE THE STATISTICS FROM ALL THE UDAS AND WATCH FOR THE TRANSFER
LIMIT TO BE EXCEEDED. AS LONG AS THE ERROR LOG IS NOT ENABLED, THE
EXCEEDING OF THE TRANSFER LIMIT WILL CAUSE THE END OF TEST #4.

EACH TIME AN ERROR OCCURS, THE "DIAGNOSTIC MACHINE" WILL TELL THE
PDP-11 PROGRAM. A MESSAGE WILL BE PRINTED (OR STORED IN THE LOG
BUFFER) AND THEN THE ERROR LIMIT FOR THE DRIVE WILL BE CHECKED. IF THE

ERROR LIMIT HAS BEEN REACHED, THE DRIVE WILL BE DROPPED FROM TESTING. IF NO MORE DRIVES REMAIN TO BE TESTED, TEST #4 WILL END (UNLESS THE ERROR LOG IS ENABLED).

WHEN THE END OF TEST #4 OCCURS, THE ACCUMULATED STATISTICS FOR EACH DRIVE WILL BE PRINTED. THIS STATISTICAL REPORT CAN BE PRINTED AT ANY TIME DURING TEST #4 BY TYPING CONTROL-C THEN THE PRINT COMMAND.

THE DATA PATTERNS TO BE USED BY TEST #4 ARE INDICATED BELOW. EACH PATTERN IS GENERATED BY WRITING THE PATTERN NUMBER IN EACH 4-BIT NIBBLE OF THE FIRST WORD, THEN REPEATING THE DATA PATTERN (SEQUENCE OF ONE TO 16 WORDS) THROUGHOUT THE REST OF THE DATA BUFFER. PATTERN NUMBER 16 WRITES NIBBLES OF ZEROS. WHEN PATTERN NUMBER ZERO IS USED, THE ACTUAL PATTERN NUMBER WRITTEN (1 TO 15) IS PLACED IN THE NIBBLES.

PATTERN 0 THIS PATTERN NUMBER IS USED TO INDICATE ANY PATTERN NUMBER 1 TO 15 CHOSEN AT RANDOM.

PATTERN 1 WORDS IN PATTERN SEQUENCE - 1

SEQUENCE (OCTAL) 105613
SEQUENCE (HEX) 8888

PATTERN 2 WORDS IN PATTERN SEQUENCE - 1

SEQUENCE (OCTAL) 031463
SEQUENCE (HEX) 3333

PATTERN 3 WORDS IN PATTERN SEQUENCE - 1

SEQUENCE (OCTAL) 030221
SEQUENCE (HEX) 3091

PATTERN 4 WORDS IN PATTERN SEQUENCE - 16 (SHIFTING ONES)

SEQUENCE (OCTAL) 000001, 000003, 000007, 000017, 000037,
000077, 000177, 000377, 000777, 001777,
003777, 007777, 017777, 037777, 077777,
177777

SEQUENCE (HEX) 0001, 0003, 0007, 000F, 001F, 003F,
007F, 00FF, 01FF, 03FF, 07FF, 0FFF,
1FFF, 3FFF, 7FFF, FFFF

PATTERN WORDS IN PATTERN SEQUENCE - 16 (SHIFTING ZEROS)

SEQUENCE (OCTAL) 177776, 177774, 177770, 177760, 177740,
177700, 177600, 177400, 177000, 176000,
174000, 170000, 160000, 140000, 100000,
000000

SEQUENCE (HEX) FFFE, FFFC, FFF8, FFF0, FFE0, FFC0,
FF80, FF00, FE00, FC00, F800, F000,
E000, C000, 8000, 0000

PATTERN 6 WORDS IN PATTERN SEQUENCE - 16

```
SEQUENCE (OCTAL)        000000, 000000, 000000, 177777, 177777,
177777, 000000, 000000, 177777, 177777,
000000, 177777, 000000, 177777, 000000,
177777

SEQUENCE (HEX)  0000, 0000, 0000, FFFF, FFFF, FFFF,
0000, 0000, FFFF, FFFF, 0000, FFFF,
0000, FFFF, 0000, FFFF

PATTERN 7       WORDS IN PATTERN SEQUENCE - (BINARY 1011011011011001)

SEQUENCE (OCTAL)        133331
SEQUENCE (HEX)    B6D9

PATTERN 8       WORDS IN PATTERN SEQUENCE - 16

SEQUENCE (OCTAL)        052525, 052525, 052525, 125252, 125252,
125252, 052525, 052525, 125252, 125252,
052525, 125252, 052525, 125252, 052525,
125252

SEQUENCE (HEX)  5555, 5555, 5555, AAAA, AAAA, AAAA,
5555, 5555, AAAA, AAAA, 5555, AAAA,
5555, AAAA, 5555, AAAA

PATTERN 9       WORDS IN PATTERN SEQUENCE - 1 (BINARY 1101101101101100)

SEQUENCE (OCTAL)        155554
SEQUENCE (HEX)    DB6C

PATTERN 10      WORDS IN PATTERN SEQUENCE - 16

SEQUENCE (OCTAL)        026455, 026455, 026455, 151322, 151322,
151322, 026455, 026455, 151322, 151322,
026455, 151322, 026455, 151322, 026455,
151322

SEQUENCE (HEX)  2D2D, 2D2D, 2D2D, D2D2, D2D2, D2D2,
2D2D, 2D2D, D2D2, D2D2, 2D2D, D2D2,
2D2D, D2D2, 2D2D, D2D2

PATTERN 11      WORDS IN PATTERN SEQUENCE - 1 (BINARY 0110110110110110)

SEQUENCE (OCTAL)        066666
SEQUENCE (HEX)    6DD6

PATTERN 12      WORDS IN PATTERN SEQUENCE - 16  (RIPPLE ONE)

SEQUENCE (OCTAL)        000001, 000002, 000004, 000010, 000020,
000040, 000100, 000200, 000400, 001000,
002000, 004000, 010000, 020000, 040000,
100000

SEQUENCE (HEX)  0001, 0002, 0004, 0008, 0010, 0020,
0040, 0080, 0100, 0200, 0400, 0800,
1000, 2000, 4000, 8000
```

PATTERN 13 WORDS IN PATTERN SEQUENCE - 16 (RIPPLE ZERO)

SEQUENCE (OCTAL) 177776, 177775, 177773, 177767, 177757,
177737, 177677, 177577, 177377, 176777,
175777, 173777, 167777, 157777, 137777,
077777

SEQUENCE (HEX) FFFE, FFFD, FFFB, FFF7, FFEF, FFDF,
FFBF, FF7F, FEFF, FDFF, FBFF, F7FF,
EFFF, DFFF, BFFF, 7FFF

PATTERN 14 WORDS IN PATTERN SEQUENCE - 3

SEQUENCE (OCTAL) 155555, 133333, 155555
SEQUENCE (HEX) DB6D, B6DB, DB6D

PATTERN 15 WORDS IN PATTERN SEQUENCE - 16

SEQUENCE (OCTAL) 133331, 133331, 133331, 155554, 155554,
155554, 133331, 133331, 155554, 155554,
133331, 155554, 133331, 155554, 133331,
155554

SEQUENCE (HEX) B6D9, B6D9, B6D9, DB6C, DB6C, DB6C,
B6D9, B6D9, DB6C, DB6C, B6D9, DB6C,
B6D9, DB6C, B6D9, DB6C

PATTERN 16 THIS IS THE OPERATOR SELECTABLE PATTERN IN MANUAL INTERVENTION MODE. QUESTIONS ARE ASKED WHEN TEST #4 IS STARTED FOR THE OPERATOR TO INPUT THE NUMBER OF WORDS IN THE SEQUENCE AND THE CONTENTS OF THE WORDS.

2

```
                                .TITLE CZUDCA0 UDA AND DRIVE DIAGNOSTIC
                                .SBTTL  PROGRAM HEADER

000000                                  .ENABL  ABS
                                        .ENABL  AMA
        002000                          .       =       2000

002000                                  BGNMOD

                                ;ASSEMBLY CONTROL

        000000                          ENG=0           ;SET NON-ZERO TO ASSEMBLE ENGINEERING CODE
        000000                          MFG=0           ;SET NON-ZERO TO ASSEMBLE MANUFACTURING CODE

                                ;++
                                ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
                                ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
                                ;--

002000                                  POINTER BGNRPT, BGNSW, BGNSFT, ERRTBL, BGNSETUP

                                        HEADER  CZDUA,A,0,0,1,PRI07
002000                                                                          L$NAME::
002000      103                                                                         .ASCII /C/
002001      132                                                                         .ASCII /Z/
002002      104                                                                         .ASCII /D/
002003      125                                                                         .ASCII /U/
002004      101                                                                         .ASCII /A/
002005      000                                                                         .BYTE   0
002006      000                                                                         .BYTE   0
002007      000                                                                         .BYTE   0
002010                                                                          L$REV::
002010      101                                                                         .ASCII  /A/
002011                                                                          L$DEPO::
002011      060                                                                         .ASCII  /0/
002012                                                                          L$UNIT::
002012  000001                                                                          .WORD   T$PTHV
002014                                                                          L$TIML::
002014  000000                                                                          .WORD   0
002016                                                                          L$HPCP::
002016  033572                                                                          .WORD   L$HARD
002020                                                                          L$SPCP::
002020  034036                                                                          .WORD   L$SOFT
002022                                                                          L$HPTP::
002022  002136                                                                          .WORD   L$HW
002024                                                                          L$SPTP::
002024  002154                                                                          .WORD   L$SW
002026                                                                          L$LADP::
002026  034622                                                                          .WORD   L$LAST
002030                                                                          L$STA::
002030  000000                                                                          .WORD   0
002032                                                                          L$CO::
002032  000000                                                                          .WORD   0
002034                                                                          L$DTYP::
002034  000001                                                                          .WORD   1
002036                                                                          L$APT::
```

```
002036  000000C
002040
002040  002124
002042
002042  000340
002044
002044  000000
002046
002046  000000
002050
002050     003
002051     003
002052
002052  000000
002054  000000
002056
002056  000000
002060
002060  002434
002062
002062  024602
002064
002064  000000
002066
002066  000000
002070
002070  000000
002072
002072  000000
002074
002074  000000
002076
002076  002460
002100
002100  104035
002102
002102  002162
002104
002104  026024
002106
002106  027476
002110
002110  027474
002112
002112  026016
002114
002114  000000
002116
002116  000000
002120
002120  000000
```

79

```
                .WORD   0
L$DTP::
                .WORD   L$DISPATCH
L$PRIO::
                .WORD   PRIO7
L$ENVI::
                .WORD   0
L$EXP1::
                .WORD   0
L$MREV::
                .BYTE   C$REVISION
                .BYTE   C$EDIT
L$EF::
                .WORD   0
                .WORD   0
L$SPC::
                .WORD   0
L$DEVP::
                .WORD   L$DVTYP
L$REPP::
                .WORD   L$RPT
L$EXP4::
                .WORD   0
L$EXP5::
                .WORD   0
L$AUT::
                .WORD   0
L$DUT::
                .WORD   0
L$LUN::
                .WORD   0
L$DESP::
                .WORD   L$DESC
L$LOAD::
                EMT     E$LOAD
L$ETP::
                .WORD   L$ERRTBL
L$ICP::
                .WORD   L$INIT
L$CCP::
                .WORD   L$CLEAN
L$ACP::
                .WORD   L$AUTO
L$PRT::
                .WORD   L$PROT
L$TEST::
                .WORD   0
L$DLY::
                .WORD   0
L$HIME::
                .WORD   0
```

DISPATCH TABLE

```
                        .SBTTL  DISPATCH TABLE

                        ;++
                        ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
                        ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
                        ;--

002122                          DISPATCH 4
002122  000004                                                          .WORD   4
002124                                                          L$DISPATCH::
002124  027504                                                          .WORD   T1
002126  030710                                                          .WORD   T2
002130  031006                                                          .WORD   T3
002132  031044                                                          .WORD   T4
```

DEFAULT HARDWARE P-TABLE

```
                                .SBTTL  DEFAULT HARDWARE P-TABLE

                                ;++
                                ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
                                ; THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
                                ; IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES,
                                ; AND IS USED AS A "TEMPLATE" FOR BUILDING THE P-TABLES.
                                ;--

002134                                  BGNHW   DFPTBL
002134  000006                                                         .WORD   L10000-L$HW/2
002136                                                          L$HW::
002136                                                          DFPTBL::

002136  172150                  .WORD   172150          ; UNIBUS ADDRESS
002140  000154                  .WORD   154             ; VECTOR ADDRESS
002142  000005                  .WORD   5.              ; BR LEVEL
002144  000000                  .WORD   0.              ; UNIBUS BURST RATE
002146  000000                  .WORD   0.              ; LOGICAL DRIVE NUMBER
002150  000000                  .WORD   0.              ; CUSTOMER DATA AREA
002152                          ENDHW
002152                                                  L10000:
```

```
                        .SBTTL  SOFTWARE P-TABLE

                        ;++
                        ; THE SOFTWARE TABLE CONTAINS VARIOUS DATA USED BY THE
                        ; PROGRAM AS OPERATIONAL PARAMETERS.  THESE PARAMETERS ARE
                        ; SET UP AT ASSEMBLY TIME AND MAY BE VARIED BY THE OPERATOR
                        ; AT RUN TIME.
                        ;--

002152                          BGNSW   SFPTBL
002152  000003                                                         .WORD   L10001-L$SW/2
002154                                                                  L$SW::
002154                                                                  SFPTBL::

                                                        ;OFFSET         USE
002154  000040                  .WORD   32.            ; 0.    ERROR LIMIT
002156  000000                  .WORD   0.             ; 2.    DATA TRANSFER LIMIT (MEGABITS)
002160  040400                  .WORD   ^B0100000100000000     ; 4.    SINGLE BIT QUESTIONS
002162                          ENDSW
002162                                                          L10001:

002162                          ENDMOD

                        .SBTTL
```

```
                                ;.TITLE GLOBAL AREAS   (3)
                                .SBTTL  GLOBAL EQUATES SECTION


002162                                  BGNMOD

                                ;++
                                ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
                                ; ARE USED IN MORE THAN ONE TEST.
                                ;--


002162                                  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
```

GLOBAL EQUATES SECTION

```
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
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                  LOE== 40000
100000                  HOE== 100000
```

GLOBAL EQUATES SECTION

```
                        ;UDA BIT DEFINITIONS
                        ;
                        ;UDASA REGISTER UNIVERSAL READ BITS
                        ;
004000                  SA.S1=  004000          ;STEP 1 STATUS BIT
010000                  SA.S2=  010000          ;STEP 2 STATUS BIT
020000                  SA.S3=  020000          ;STEP 3 STATUS BIT
040000                  SA.S4=  040000          ;STEP 4 STATUS BIT
100000                  SA.ERR= 100000          ;ERROR INDICATOR
                        ;
                        ;UDASA REGISTER ERROR STATUS BITS
                        ;
003777                  SA.ERC= 003777          ;ERROR CODE
                        ;
                        ;UDASA REGISTER STEP ONE READ BITS
                        ;
002000                  SA.NV=  002000          ;NON SETTABLE INTERRUPT VECTOR
001000                  SA.A2=  001000          ;22 BIT ADDRESS BUS
000400                  SA.DI=  000400          ;ENHANCED DIAGNOSTICS
                        ;       000377          ;ALL BITS RESERVED
                        ;
                        ;UDASA REGISTER STEP ONE WRITE BITS
                        ;
000177                  SA.VEC= 000177          ;INTERRUPT VECTOR (DIVIDED BY 4)
000200                  SA.INT= 000200          ;INTERRUPT ENABLE DURING INITIALIZATION
003400                  SA.MSG= 003400          ;MESSAGE RING LENGTH
034000                  SA.CMD= 034000          ;COMMAND RING LENGTH
                        ;       040000          ;RESERVED
100000                  SA.STP= 100000          ;STEP - MUST ALWAYS BE WRITTEN A ONE
                        ;
                        ;UDASA REGISTER STEP TWO READ BITS
                        ;
000007                  SA.MSE= 000007          ;MESSAGE RING LENGTH ECHO
000070                  SA.CME= 000070          ;COMMAND RING LENGTH ECHO
                        ;       000100          ;RESERVED
000200                  SA.STE= 000200          ;STEP ECHO
003400                  SA.CTP= 003400          ;CONTROLLER TYPE
                        ;
                        ;UDASA REGISTER STEP TWO WRITE BITS
                        ;
000001                  SA.PRG= 000001          ;ENABLE VAX UNIBUS ADAPTER PURGE INTERRUPT
                        ;       177776          ;LOW ORDER MESSAGE RING BYTE ADDRESS
                        ;
                        ;UDASA REGISTER STEP THREE RFAD BITS
                        ;
000177                  SA.VCE= 000177          ;INTERRUPT VECTOR ECHO
000200                  SA.INE= 000200          ;INTERRUPT ENABLE ECHO
000400                  SA.NVE= 000400          ;VECTOR NOT PROGRAMMABLE
                        ;       003000          ;RESERVED
                        ;
                        ;UDASA REGISTER STEP THREE WRITE BITS
                        ;
                        ;       077777          ;HIGH ORDER MESSAGE RING BYTE ADDRESS
100000                  SA.TST= 100000          ;PURGE POLE TEST ENABLE
                        ;
                        ;UDASA REGISTER STEP FOUR READ BITS
                        ;
```

```
     58          000377                  SA.MCV= 000377                  ;UDA MICROCODE VERSION
     59                                  ;       003400                  ;RESERVED
     60
     61                                  ;UDASA REGISTER STEP FOUR WRITE BITS
     62
     63          000001                  SA.GO=  000001                  ;GO BIT TO START UDA FIRMWARE
     64          000002                  SA.LFC= 000002                  ;LAST FAILURE CODE REQUEST
     65          000374                  SA.BST= 000374                  ;BURST LEVEL
```

GLOBAL EQUATES SECTION

```
                                ;COMMAND/MESSAGE DESCRIPTOR BIT DEFINITIONS

100000                          RG.OWN= 100000                  ;SET WHEN UDA OWNS RING
040000                          RG.FLG= 040000                  ;FLAG BIT

                                ;OFFSETS INTO HOST COMMUNICATIONS AREA WITH ONE DESCRIPTOR TO EACH RING
                                ;AND TWO PACKET AND BUFFER AREAS.

000004                          HC.ISZ=  4.                     ;SIZE OF INTERRUPT INDICATOR WORDS
000004                          HC.RSZ=  4.                     ;SIZE OF RING IN BYTES
000004                          HC.ESZ=  4.                     ;SIZE OF ENVELOPE WORDS BEFORE PACKET
000060                          HC.PSZ= 48.                     ;SIZE OF COMMAND AND MESSAGE PACKETS
000074                          HC.BSZ= 60.                     ;SIZE OF BUFFER

000000                          HC.INT=  0.                     ;INTERRUPT INDICATOR WORDS START
000004                          HC.MSG= HC.INT+HC.ISZ           ;MESSAGE RING START
000006                          HC.MCT= HC.MSG+2.               ;MESSAGE RING CONTROL WORD
000010                          HC.CMD= HC.MSG+HC.RSZ           ;COMMAND RING START
000012                          HC.CCT= HC.CMD+2.               ;COMMAND RING CONTROL WORDS
000014                          HC.MEV= HC.CMD+HC.RSZ           ;MESSAGE ENVELOPE START
000020                          HC.MPK= HC.MEV+HC.ESZ           ;MESSAGE PACKET START
000100                          HC.CEV= HC.MPK+HC.PSZ           ;COMMAND ENVELOPE START
000104                          HC.CPK= HC.CEV+HC.ESZ           ;COMMAND PACKET START
000164                          HC.BF1= HC.CPK+HC.PSZ           ;FIRST BUFFER
000260                          HC.BF2= HC.BF1+HC.BSZ           ;SECOND BUFFER

000354                          HC.SIZ= HC.BF2+HC.BSZ           ;TOTAL SIZE OF HOST COMM AREA

                                ;VIRTUAL CIRCUIT IDENTIFIERS

000000                          MSCP=     0                     ;MSCP CIRCUIT
000001                          LOG=      1                     ;LOG CIRCUIT
177777                          DIAG=    -1                     ;DIAGNOSTIC CIRCUIT
001000                          DUP=     1000                   ;DIAGNOSTIC AND UTILITIES PROTOCOL
```

```
;                 +----------------------------------------------+
;       HC.INT    !        INTERRUPT INDICATORS                  !    4 BYTES
;                 !                                              !
;                 +----------------------------------------------+
;       HC.MSG    .        MESSAGE RING                          !    4 BYTES
;       HC.MCT    !                                              !
;                 +----------------------------------------------+
;       HC.CMD    :        COMMAND RING                          !    4 BYTES
;       HC.CCT    !                                              !
;                 +----------------------------------------------+
;       HC.MEV    !        MESSAGE ENVELOPE                      !   52 BYTES
;       HC.MPK    :                                              !
;                 .                                              !
;                                                                !
;                                                                !
;                 !                                              !
;                 +----------------------------------------------+
;       HC.CEV    .        COMMAND ENVELOPE                      !   52 BYTES
;       HC.CPK    .                                              !
;                 .                                              !
;                                                                !
;                 .                                              !
;                 !                                              !
;                 +----------------------------------------------+
;       HC.BF1    !        BUFFER # 1 (RESPONSE TO DM PROGRAM)   !   60 BYTES
;                 !                                              !
;                 !                                              !
;                 !                                              !
;                 +----------------------------------------------+
;       HC.BF2    .        BUFFER # 2 (REQUEST FROM DM PROGRAM)  !   60 BYTES
;                 .                                              !
;                 .                                              !
;                 .                                              !
;                 +----------------------------------------------+
```

GLOBAL EQUATES SECTION

```
                        ;COMMAND PACKET OPCODES

000001          OP.ABO=   1           ;ABORT COMMAND
000020          OP.ACC=  20           ;ACCESS COMMAND
000010          OP.AVL=  10           ;AVAILABLE COMMAND
000021          OP.CCD=  21           ;COMPARE CONTROLLER DATA COMMAND
000040          OP.CMP=  40           ;COMPARE HOST DATA COMMAND
000022          OP.ERS=  22           ;ERASE COMMAND
000023          OP.FLU=  23           ;FLUSH COMMAND
000002          OP.GCS=   2           ;GET COMMAND STATUS COMMAND
000003          OP.GUS=   3           ;GET UNIT STATUS COMMAND
000011          OP.ONL=  11           ;ONLINE COMMAND
000041          OP.RD=   41           ;READ COMMAND
000024          OP.RPL-  24           ;REPLACE COMMAND
000004          OP.SCC=   4           ;SET CONTROLLER CHARACTERISTICS COMMAND
000012          OP.SUC-  12           ;SET UNIT CHARACTERISTICS COMMAND
000042          OP.WR-   42           ;WRITE COMMAND
000030          OP.MRD=  30           ;MAINTENANCE READ COMMAND
000031          OP.MWR-  31           ;MAINTENANCE WRITE COMMAND
000200          OP.END= 200           ;END PACKET FLAG
000007          OP.SEX=   7           ;SERIOUS EXCEPTION END PACKET
000100          OP.AVA- 100           ;AVAILABLE ATTENTION MESSAGE
000101          OP.DUP= 101           ;DUPLICATE UNIT NUMBER ATTENTION MESSAGE
000102          OP.SHC- 102           ;SHADOW COPY COMPLETE ATTENION MESSAGE
000103          OP.RLC= 103           ;RESET COMMAND LIMIT ATTENTION MESSAGE

000001          OP.GSS=   1           ;DUP GET STUD STATUS
000002          OP.ESP=   2           ;DUP EXECUTE SUPPLIED PROGRAM
000003          OP.ELP=   3           ;DUP EXECUTE LOCAL PROGRAM
000004          OP.SSD-   4           ;DUP SEND STUD DATA
000005          OP.RSD-   5           ;DUP RECEIVE STUD DATA

                        ;NOTE: END PACKET OPCODES (ALSO CALLED ENDCODES) ARE FORMED BY ADDING THE END
                        ;PACKET FLAG TO THE COMMAND OPCODE. FOE EXAMPLE, A READ COMMAND'S END PACKE-
                        ;CONTAINS THE VALUE OP.RD+OP.END IN ITS OPCODE FIELD. THE INVALID COMMAND END
                        ;PACKET CONTAINS JUST THE END PACKET FLAG (I.E., OP.END) IN ITS OPCODE FIELD.
                        ;THE SERIOUS EXCEPTION END PACKET CONTAINS THE SUM OF THE END PACKET FLAG
                        ;PLUS THE SERIOUS EXCEPTION OPCODE SHOWN ABOVE (I.E., OP.SEX+OP.END) IN ITS
                        ;OPCODE FIELD.
                        ;
                        ;COMMAND OPCODE BITS 3 THROUGH 5 INDICATE THE COMMAND CLASS, WHICH IS ENCODED
                        ;AS FOLLOWS:
                        ;    000  IMMEDIATE COMMANDS
                        ;    001  SEQUENTIAL COMMANDS
                        ;    010  NON-SEQUENTIAL COMMANDS THAT DO NOT INCLUDE A BUFFER DESCRIPTOR
                        ;    100  NON-SEQUENTIAL COMMANDS THAT DO INCLUDE A BUFFER DESCRIPTOR
```

GLOBAL EQUATES SECTION

```
                        ;COMMAND MODIFIERS
                        ;       = 020000                ;CLEAR SERIOUS EXCEPTION
040000                  MD.CMP= 040000                  ;COMPARE
100000                  MD.EXP= 100000                  ;EXPRESS REQUEST
010000                  MD.ERR= 010000                  ;FORCE ERROR
004000                  MD.SCH= 004000                  ;SUPPRESS CACHING (HIGH SPEED)
002000                  MD.SCL= 002000                  ;SUPPRESS CACHING (LOW SPEED)
000100                  MD.SEC= 000100                  ;SUPPRESS ERROR CORRECTION
000400                  MD.SER= 000400                  ;SUPPRESS ERROR RECOVERY
000200                  MD.SSH= 000200                  ;SUPPRESS SHADOWING
000100                  MD.WBN= 000100                  ;WRITE-BACK (NON-VOLATILE)
000400                  MD.WBV= 000400                  ;WRITE BACK (VOLATILE)
000020                  MD.SEQ= 000020                  ;WRITE SHADOW SET ONE UNIT AT A TIME
000001                  MD.SPD= 000001                  ;SPIN-DOWN
000001                  MD.FEU= 000001                  ;FLUSH ENTIRE UNIT
000002                  MD.VOL= 000002                  ;VOLATILE ONLY
000001                  MD.NXU= 000001                  ;NEXT UNIT
000001                  MD.RIP= 000001                  ;ALLOW SELF DESTRUCTION
000002                  MD.IMF= 000002                  ;IGNORE MEDIA FORMAT ERROR
000004                  MD.SWP= 000004                  ;SET WRITE PROTECT
000010                  MD.CWB= 000010                  ;CLEAR WRITE-BACK DATA LOST
000001                  MD.PRI= 000001                  ;PRIMARY REPLACEMENT BLOCK

                        ;END PACKET FLAGS

000200                  EF.BBR= 000200                  ;BAD BLOCK REPORTED
000100                  EF.BBU= 000100                  ;BAD BLOCK UNREPORTED
000040                  EF.LOG= 000040                  ;ERROR LOG GENERATED
000020                  EF.SEX= 000020                  ;SERIOUS EXCEPTION

                        ;CONTROLLER FLAGS

000200                  CF.ATN= 000200                  ;ENABLE ATTENTION MESSAGES
000100                  CF.MSC= 000100                  ;ENABLE MISCELLANEOUS ERROR LOG MESSAGES
000040                  CF.OTH= 000040                  ;ENABLE OTHER HOST'S ERROR LOG MESSAGES
000020                  CF.THS= 000020                  ;ENABLE THIS HOST'S ERROR LOG MESSAGES
000002                  CF.SHD= 000002                  ;SHADOWING
000001                  CF.576= 000001                  ;576 BYTE SECTORS
```

```
                                ;UNIT FLAGS
                                
000001                          UF.CMR= 000001                  ;COMPARE READS
000002                          UF.CMW= 000002                  ;COMPARE WRITES
100000                          UF.RPL= 100000                  ;HOST INITIATED BAD BLOCK REPLACEMENT
040000                          UF.INA= 040000                  ;INACTIVE SHADOW SET UNIT
004000                          UF.SCH= 004000                  ;SUPPRESS CACHING (HIGH SPEED)
002000                          UF.SCL= 002000                  ;SUPPRESS CACHING (LOW SPEED)
000100                          UF.WBN= 000100                  ;WRITE-BACK (NON-VOLATILE)
020000                          UF.WPH= 020000                  ;WRITE PROTECT (HARDWARE)
001000                          UF.WPS= 001000                  ;WRITE PROTECT (SOFTWARE OR VOLUME)
000004                          UF.576= 000004                  ;576 BYTE SECTORS

                                ;COMMAND PACKET OFFSETS

                                ;                       GENERIC COMMAND PACKET OFFSETS:
000000                          P.CRF=   0.                     ;COMMAND REFERENCE NUMBER
000004                          P.UNIT=  4.                     ;UNIT NUMBER
000010                          P.OPCD=  8.                     ;OPCODE
000012                          P.MOD=  10.                     ;MODIFIERS
000014                          P.BCNT= 12.                     ;BYTE COUNT
000020                          P.BUFF= 16.                     ;BUFFER DESCRIPTOR
000020                          P.UADR= 16.                     ;UNIBUS ADDRESS OF BUFFER DESCRIPTOR
000034                          P.LBN=  28.                     ;LOGICAL BLOCK NUMBER

                                ;                       ABORT AND GET COMMAND STATUS COMMAND PACKET OFFSETS:
000014                          P.OTRF= 12.                     ;OUTSTANDING REFERENCE NUMBER

                                ;                       ONLINE AND SET UNIT CHARACTERISTICS COMMAND PACKET OFFSETS:
000016                          P.UNFL= 14.                     ;UNIT FLAGS
000020                          P.HSTI= 16.                     ;HOST IDENTIFIER / RESERVED
000034                          P.ELGF= 28.                     ;ERROR LOG FLAGS
000040                          P.SHUN= 32.                     ;SHADOW UNIT
000042                          P.CPSP= 34.                     ;COPY SPEED
                                ;                       REPLACE COMMAND PACKET OFFSETS:
000014                          P.RBN=  12.                     ;REPLACEMENT BLOCK NUMBER

                                ;                       SET CONTROLLER CHARACTERISTICS COMMAND PACKET OFFSETS:
000014                          P.VRSN= 12.                     ;MSCP VERSION
000016                          P.CNTF= 14.                     ;CONTROLLER FLAGS
000020                          P.HTMO= 16.                     ;HOST TIMEOUT
000022                          P.USEF  18.                     ;USE FRACTION
000024                          P.TIME= 20.                     ;QUAD-WORD TIME AND DATE

                                ;                       MAINTENANCE READ AND MAINTENANCE WRITE COMMAND PACKET OFFSETS:
000034                          P.RGID= 28.                     ;REGION ID
000040                          P.RGOF= 32.                     ;REGION OFFSET

                                ;                       EXECUTE SUPPLIED PROGRAM COMMAND PACKET OFFSETS:
000024                          P.DMDT= 20.                     ;DMDT TERMINAL ADDRESS (MAINT WRITE ONLY)
000034                          P.OVRL- 28.                     ;BUFFER DESCRIPTOR FOR OPERLAYS
```

GLOBAL EQUATES SECTION

```
                        ;END PACKET OFFSETS
                        ;
                        ;               GENERIC END PACKET OFFSETS:
000000                  P.CRF=   0.                     ;COMMAND REFERENCE NUMBER
000004                  P.UNIT=  4.                     ;UNIT NUMBER
000010                  P.OPCD=  8.                     ;OPCODE (ALSO CALLED ENDCODE)
000011                  P.FLGS=  9.                     ;END PACKET FLAGS
000012                  P.STS=  10.                     ;STATUS
000014                  P.BCNT= 12.                     ;BYTE COUNT
000034                  P.FBBK= 28.                     ;FIRST BAD BLOCK
                        ;
                        ;               GET COMMAND STATUS END PACKET OFFSETS:
000014                  P.OTRF= 12.                     ;OUTSTANDING REFERENCE NUMBER
000020                  P.CMST= 16.                     ;COMMAND STATUS
                        ;
                        ;               GET UNIT STATUS END PACKET OFFSETS:
000014                  P.MLUN= 12.                     ;MULTI-UNIT CODE
000016                  P.UNFL= 14.                     ;UNIT FLAGS
000020                  P.HSTI= 16.                     ;HOST IDENTIFIER
000024                  P.UNTI= 20.                     ;UNIT IDENTIFIER
000034                  P.MEDI= 28.                     ;MEDIA TYPE IDENTIFIER
000040                  P.SHUN= 32.                     ;SHADOW UNIT
000042                  P.SHST= 34.                     ;SHADOW STATUS
000044                  P.TRCK= 36.                     ;TRACK SIZE
000046                  P.GRP=  38.                     ;GROUP SIZE
000050                  P.CYL=  40.                     ;CYLINDER SIZE
000054                  P.RCTS= 44.                     ;RCT TABLE SIZE
000056                  P.RBNS= 46.                     ;RBNS / TRACK
000057                  P.RCTC= 47.                     ;RCT COPIES
                        ;
                        ;               ONLINE AND SET UNIT CHARACTERISTICS END PACKET AND AVAILABLE
                        ;               ATTENTION MESSAGE OFFSETS:
000014                  P.MLUN= 12.                     ;MULTI-UNIT CODE
000016                  P.UNFL= 14.                     ;UNIT FLAGS
000020                  P.HSTI= 16.                     ;HOST IDENTIFIER
000024                  P.UNTI= 20.                     ;UNIT IDENTIFIER
000034                  P.MEDI= 28.                     ;MEDIA TYPE IDENTIFIER
000040                  P.SHUN= 32.                     ;SHADOW UNIT
000042                  P.SHST= 34.                     ;SHADOW STATUS
000044                  P.UNCL= 36.                     ;UNIT COMMAND LIMIT
000050                  P.UNSZ= 40.                     ;UNIT SIZE
000054                  P.VSER= 44.                     ;VOLUME SERIAL NUMBER
                        ;
                        ;               SET CONTROLLER CHARACTERISTICS END PACKET OFFSETS:
000014                  P.VRSN= 12.                     ;MSCP VERSION
000016                  P.CNTF= 14.                     ;CONTROLLER FLAGS
000020                  P.CTMO= 16.                     ;CONTROLLER TIMEOUT
000022                  P.CNCL= 18.                     ;CONTROLLER COMMAND LIMIT
000024                  P.CNTI= 20.                     ;CONTROLLER ID
```

GLOBAL EQUATES SECTION

```
                                ;STATUS AND EVENT CODE DEFINITIONS

000037                          ST.MSK= 37              ;STATUS / EVENT CODE MASK
000040                          ST.SUB= 40              ;SUB-CODE MULTIPLIER
000000                          ST.SUC=  0              ;SUCCESS
000001                          ST.CMD=  1              ;INVALID COMMAND
000002                          ST.ABO=  2              ;COMMAND ABORTED
000003                          ST.OFL=  3              ;UNIT-OFFLINE
000004                          ST.AVL=  4              ;UNIT-AVAILABLE
000005                          ST.MFE=  5              ;MEDIA FORMAT ERROR
000006                          ST.WPR=  6              ;WRITE PROTECTED
000007                          ST.CMP=  7              ;COMPARE ERROR
000010                          ST.DAT= 10              ;DATA ERROR
000011                          ST.HST= 11              ;HOST BUFFER ACCESS ERROR
000012                          ST.CNT= 12              ;CONTROLLER ERROR
000013                          ST.DRV= 13              ;DRIVE ERROR
000037                          ST.DIA= 37              ;MESSAGE FROM AN INTERNAL DIAGNOSTIC
```

```
                                ;CONTROLLER TABLE DEFINITIONS
                                ;
                                ;ONE TABLE WILL BE SET UP BY INITIALIZE SECTION FOR EACH UDA SELECTED
                                ;FOR TESTING.  TABLES ARE CONTIGUOUS.  THE END OF THE TABLES IS
                                ;MARKED BY A WORD OF ZEROS.
                                ;
                                ;THE FIRST TABLE IS POINTED TO BY THE CONTENTS OF CTABS.
                                ;THE NUMBER OF TABLES IS CONTAINED IN CTRLRS.

000000                          C.UADR=  0.                             ;UNIBUS ADDRESS OF UDAIP REGISTER
000002                          C.UNIT=  2.
000077                                  CT.UNT= 000077                  ; LOGICAL UNIT NUMBER (FIRST)
100000                                  CT.AVL= BIT15                   ; SET WHEN NOT AVAILABLE FOR TEST
000004                          C.VEC=   4.
000777                                  CT.VEC= 000777                  ; VECTOR ADDRESS
007000                                  CT.BRL= 007000                  ; BR LEVEL
000006                          C.BST=   6.                             ; BURST LEVEL
000010                          C.JSR=   8.                             ;INTERRUPT SERVICE ROUTINE FOR CONTROLLER
000012                          C.JAD=  10.                             ;   THESE TWO WORDS LOADED WITH [JSR R0,UDASRV]
000014                          C.FLG=  12.                             ;FLAGS
000002                                  CT.RN=  BIT1                    ;DM PROGRAM RUNNING
000004                                  CT.CMD= BIT2                    ;COMMAND ISSUED, WAITING FOR RESPONSE
000010                                  CT.MSG= BIT3                    ;MESSAGE RESPONSE RECEIVED
                                                                        ;WHENEVER THIS BIT IS SET, CT.CMD IS CLEARED
000020                                  CT.REQ= BIT4                    ;BUFFER HAS BEEN GIVEN TO UDA FOR REQUEST
                                                                        ;SET WHENEVER READ STUD DATA COMMAND
                                                                        ;GIVEN TO UDA
000016                          C.RING= 14.                             ;RING BUFFER ADDRESS
000020                          C.DR0=  16.                             ;POINTER TO DRIVE TABLES
000022                          C.DR1=  18.                             ;   IF ZERO, NO DRIVE TABLE EXISTS
000024                          C.DR2=  20.
000026                          C.DR3=  22.
000030                          C.DR4=  24.
000032                          C.DR5=  26.
000034                          C.DR6=  28.
000036                          C.DR7=  30.
000040                          C.TO=   32.                             ;TIMEOUT COUNTER
000042                          C.TOH=  34.                             ;   (TWO WORDS)
000044                          C.REF=  36.                             ;COMMAND REFERENCE NUMBER

000046                          C.SIZE= 38.                             ;SIZE OF CONTROLLER TABLE IN BYTES
```

```
                                ;DRIVE TABLE DEFINITIONS
                                ;
                                ;ONE DRIVE TABLE WILL BE SET UP BY THE INITIALIZE SECTION FOR EACH
                                ;DRIVE SELECTED FOR TESTING.  EACH TABLE IS POINTED T  BY A
                                ;WORD IN THE CONTROLLER TABLE ON WHICH THE DRIVE EXISTS.
                                ;
000000                          D.DRV=   0.                             ;DRIVE NUMBER
000002                          D.UNIT  =       D.DRV+2
000077                                  DT.UNT= 000077                  ; LOGICAL UNIT NUMBER OF DRIVE
100000                                  DT.AVL= BIT15                   ; SET WHEN NOT AVAILABLE FOR TESTING
000004                          D.PRM   =       D.UNIT+2                ;HARDWARE QUESTION FLAGS
040000                                  D.IW    =BIT14                  ;INITIAL WRITE
020000                                  D.DCY   =BIT13                  ;DIAGNOSTIC CYLINDERS
010000                                  D.ECC   =BIT12                  ;ECC CORRECTION ENABLED
004000                                  D.RO    =BIT11                  ;READ ONLY
002000                                  D.WO    =BIT10                  ;WRITE ONLY
001000                                  D.RET   =BIT9                   ;RETRIES ENABLED
000400                                  D.CYL   =BIT8                   ;START/END CYLINDERS SPECIFIED
000100                                  D.SEQ   =BIT6                   ;SEQUENTIAL ACCESS
000040                                  D.BE    =BIT5                   ;BEGIN/END BLOCKS USED
000020                                  D.TR    =BIT4                   ;WHEN D.BE=0: 1 - TRACKS, 0 - GROUPS
000010                                  D.WC    =BIT3                   ;WRITE CHECKS ENABLED
000004                                  D.WCA   =BIT2                   ;ALWAYS WRITE CHECK
000002                                  D.DC    =BIT1                   ;DATA COMPARES ENABLED
000001                                  D.DCA   =BIT0                   ;ALWAYS DATA COMPARE
011012                                  DDEF=D.ECC+D.WC+D.DC+D.RET      ;DEFAULT D.PRM
140200                                  D.ZERO=BIT15+BIT7+D.IW          ;BITS TO BE CLEARED
000006                          D.PAT   -       D.PRM+2                 ;DATA PATTERN NUMBER
000010                          D.BB    =       D.PAT+2                 ;BAD BLOCK COUNT
000012                          D.BB01  =       D.BB+2                  ;BAD BLOCK 1
000016                          D.BB02  =       D.BB01+4                ;          2
000022                          D.BB03  -       D.BB02+4                ;          3
000026                          D.BB04  =       D.BB03+4                ;          4
000032                          D.BB05  =       D.BB04+4                ;          5
000036                          D.BB06  =       D.BB05+4                ;          6
000042                          D.BB07  =       D.BB06+4                ;          7
000046                          D.BB08  =       D.BB07+4                ;          8
000052                          D.BB09  =       D.BB08+4                ;          9
000056                          D.BB10  =       D.BB09+4                ;         10
000062                          D.BB11  =       D.BB10+4                ;         11
000066                          D.BB12  =       D.BB11+4                ;         12
000072                          D.BB13  =       D.BB12+4                ;         13
000076                          D.BB14  -       D.BB13+4                ;         14
000102                          D.BB15  -       D.BB14+4                ;         15
000106                          D.BB16  -       D.BB15+4                ;         16
000112                          D.BEC   =       D.BB16+4                ;BEGIN/END SET COUNT
000114                          D.BGN1  =       D.BEC+2                 ;BEGIN BLOCK 1
000120                          D.END1  =       D.BGN1+4                ;END
000124                          D.BGN2  =       D.END1+4                ;BEGIN BLOCK 2
000130                          D.END2  =       D.BGN2+4                ;END
000134                          D.BGN3  =       D.END2+4                ;BEGIN BLOCK 3
000140                          D.END3  =       D.BGN3+4                ;END
000144                          D.BGN4  =       D.END3+4                ;BEGIN BLOCK 4
000150                          D.END4  =       D.BGN4+4                ;END
000154                          D.BCYL  =       D.END4+4                ;BEGIN CYLINDER
000160                          D.ECYL  =       D.BCYL+4                ;END CYLINDER
000164                          D.XFRW  -       D.ECYL+4                ;MEGABITS WRITTEN COUNT
```

GLOBAL EQUATES SECTION

```
     58          000166                D.XFRR  =       D.XFRW+2        ;MEGABITS READ COUNT
     59          000170                D.HERR  =       D.XFRR+2        ;HARD ERROR COUNTER
     60          000172                D.SERR  =       D.HERR+2        ;SOFT ERROR COUNTER
     61          000174                D.SEEK  =       D.SERR+2        ; NUMBER OF SEEKS X1000
     62          000176                D.ECCC  =       D.SEEK+2        ;ECC COUNTER
     63          000200                D.SERN  =       D.ECCC+2        ;DRIVE SERIAL NUMBER
     64
     65          000206                D.SIZE  =       D.SERN+6        ;SIZE OF DRIVE TABLE IN BYTES
```

GLOBAL EQUATES SECTION

```
                        ;KT MEMORY MANAGEMENT REGISTERS
                        
172340                  PAR0=   172340          ;KERNAL PAGE ADDRESS REGISTERS
172342                  PAR1=   172342
172344                  PAR2=   172344
172346                  PAR3=   172346
172350                  PAR4=   172350
172352                  PAR5=   172352
172354                  PAR6=   172354
172356                  PAR7=   172356

172300                  PDR0=   172300          ;KERNAL PAGE DESCRIPTOR REGISTERS
172302                  PDR1=   172302
172304                  PDR2=   172304
172306                  PDR3=   172306
172310                  PDR4=   172310
172312                  PDR5=   172312
172314                  PDR6=   172314
172316                  PDR7=   172316
077400                  MM.PLF= 077400          ;PAGE LENGTH FIELD
                                                ;NORMALLY ALL SET
000100                  MM.W=   000100          ;WRITTEN INTO (READ ONLY)
000010                  MM.ED=  000010          ;EXPANSION DIRECTION
                                                ;NORMALLY NOT SET
000006                  MM.ACF= 000006          ;ACCESS CONTROL FIELD
                                                ;NORMALLY ALL SET

177572                  SR0=    177572          ;STATUS REGISTER 0
000001                  MM.EN=  000001          ;ENABLE MANAGEMENT


                        ;DM PROGRAM HEADER DEFINITIONS

000000                  DMTRLN= 0               ;OFFSET TO SIZE OF PROGRAM NEEDING DOWNLINE LOAD
000004                  DMOVRL= 4               ;OFFSET TO SIZE OF OVERLAY
000040                  DMMAIN= 40              ;OFFSET TO FIRST WORD OF MAIN PROGRAM
001000                  DMFRST= 1000            ;ADDRESS IN DM FILE CONTAINING FIRST BYTE OF HEADER
```

```
                        ;USEFUL INSTRUCTION DEFINITIONS

                        .MACRO  AND ARG,ADR                     ;LOGICAL AND INSTRUCTION
                                .LIST
                                        BIC #^C<ARG>,ADR
                                .NLIST
                        .ENDM

                        .MACRO  OR ARG,ADR                      ;LOGICAL OR INSTRUCTION
                                .LIST
                                        BIS #ARG,ADR
                                .NLIST
                        .ENDM

                        .MACRO  PUSH ARG                        ;PUSH INSTRUCTION
                                .IRP X,<ARG>
                                .LIST
                                        MOV X,-(SP)
                                .NLIST
                                .ENDM
                        .ENDM

                        .MACRO  POP ARG                         ;POP INSTRUCTION
                                .IRP X,<ARG>
                                .LIST
                                        MOV (SP)+,X
                                .NLIST
                                .ENDM
                        .ENDM

                        .MACRO  .BR ADR                                 ;A BRANCH TO THE NEXT LOCATION
                                .IF P2
                                    .IF ADR NE .
                                        .ERROR ;ILLEGAL .BR TO ADR
                                    .ENDC
                                .ENDC
                        .ENDM

                        .MACRO  ASSUME FIRST CONDITION SECOND
                                    .IF CONDITION <FIRST>-<SECOND>
                                    .IFF
                                        .ERROR ;BAD ASSUME OF <FIRST> CONDITION <SECOND>
                                    .ENDC
                        .ENDM
```

GLOBAL DATA SECTION

```
                                .SBTTL  GLOBAL DATA SECTION

                                ;++
                                ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
                                ; IN MORE THAN ONE TEST.
                                ;--


                                        ERRTBL
                                                                        L$ERRTBL::
002162  000000                  ERRTYP::      .WORD   0
002164  000000                  ERRNBR::      .WORD   0
002166  000000                  ERRMSG::      .WORD   0
002170  000000                  ERRBLK::      .WORD   0

002172      0C0     000         ERRCHR: .BYTE 0,0                     ;FIRST BYTE LOADED WITH OUTPUT CHARACTER
                                                                      ;SECOND BYTE REMAINS ZERO TO STOP OUTPUT
002174                          FFREE:: .BLKW 1                       ;FIRST FREE WORD IN MEMORY
002176                          FSIZE:: .BLKW 1                       ;SIZE OF FREE MEMORY IN WORDS
002200                          FMEM:   .BLKW 1                       ;COPY OF FFREE AT END OF INIT SECTION
002202                          FMEMS:  .BLKW 1                       ;COPY OF FSIZE AT END OF INIT SECTION
002204                          CTABS:: .BLKW 1                       ;START OF CONTROLLER TABLE STORAGE
002206                          CTRLRS: .BLKW 1                       ;COUNT OF UDA CONTROLLERS IN PTABLES
002210                          TSTTAB: .BLKW 1                       ;POINTER TO FIRST CONTROLLER TABLE UNDER TEST
002212                          DMPROG: .BLKW 1                       ;START ADDRESS OF DM PROGRAM
002214                          DMEND:  .BLKW 1                       ;END ADDRESS OF DM PROGRAM(FIRST FREE MEMORY ADR)
002216                          DMENDS: .BLKW 1                       ;FREE MEMORY SIZE FROM END OF DM PROGRAM

002220                          KTBASA: .BLKW 1                       ;HIGH TWO BYTES OF BASE ADDRESS FOR KT ACCESS
002222                          KTBASO: .BLKW 1                       ;LOW BYTE OF ADDRESS FOR KT ACCESS

002224                          IFLAGS::.BLKW 1                       ;FLAGS FROM INIT CODE FOR TEST 4
        000002                                  ICONT ==BIT1          ; CONTINUE EVENT FLAG
        000004                                  IREST ==BIT2          ; RESTART FLAG
        000010                                  ISTRT ==BIT3          ; START FLAG
        000020                                  ISTRTH==BIT4          ; START FLAG HOLD FOR T4UPRM ROUTINE
002226                          TNUM:   .BLKW 1                       ;NUMBER OF TEST EXECUTING
002230                          URUN:   .BLKW 1                       ;NUMBER OF UNITS TO RUN AT ONE TIME
002232                          URNING: .BLKW 1                       ;NUMBER OF UNITS STILL RUNNING
002234                          UCNT:   .BLKW 1                       ;COUNTER OF UNITS UNDER TEST
002236                          INTRCV: .BLKW 1                       ;INTERRUPT RECEIVED FLAG FOR INT TESTING

002240                          FNAME:
002240      132     125     104         .ASCIZ\ZUDDA0.PAK\            ;NAME OF DATA FILE
                                        .EVEN
002254  000000                  FDATA:  .WORD 0
002256  000000                  FILOPN: .WORD 0                       ;FILE OPEN WHEN NON-ZERO
002260                          TEMP:   .BLKW 12.                     ;TEMPORY STORAGE FOR GMANI RESPONSES

002310  000001                  PAT16C: .WORD 1                       ;COUNT OF WORDS IN DATA PATTERN 16
002312  000000                  PAT16W: .WORD 0                       ;WORD SEQUENCE FOR DATA PATTERN 16
002314  000000                          .WORD 0
002316  000000                          .WORD 0
002320  000000                          .WORD 0
002322  000000                          .WORD 0
002324  000000                          .WORD 0
002326  000000                          .WORD 0
```

GLOBAL DATA SECTION

```
     68 002330  000000                          .WORD 0
     69 002332  000000                          .WORD 0
     70 002334  000000                          .WORD 0
     71 002336  000000                          .WORD 0
     72 002340  000000                          .WORD 0
     73 002342  000000                          .WORD 0
     74 002344  000000                          .WORD 0
     75 002346  000000                          .WORD 0
     76 002350  000000                          .WORD 0
```

GLOBAL DATA SECTION

```
                                        ;CLOCK CONTROL
                                        
002352  000000                  KW.CSR: .WORD 0                 ;CSR OF CLOCK
002354  000000                  KW.BRL: .WORD 0                 ;BR LEVEL
002356  000000                  KW.VEC: .WORD 0                 ;VECTOR
002360  000000                  KW.HZ:  .WORD 0                 ;HERTZ (50. OR 60.)
002362                          KW.EL:  .BLKW 2                 ;ELAPSED TIME
                                        
002366  000000                  NXMAD:  .WORD 0                 ;SET TO ALL ONES BY NON-EXISTANT ADDRESS
002370  000000                  KTMEM:  .WORD 0                 ;SET TO ALL ONES IF NO KT EXISTS
                                        
002372                          STIME:  .BLKW 2                 ;STATISTICAL REPORT TIMER
                                        
002376                          T2WRR:  .BLKW 1                 ;WRITE/READ REGION
002400                          T2WRO:  .BLKW 1                 ;WRITE/READ OFFSET
002402                          T2DR:   .BLKW 1                 ;DIAGNOSE REGION
                                        
                                        
                                        ;ERROR LOG CONTROL WORDS
                                        
002404                          LBUFS:  .BLKW 1                 ;START ADDRESS OF LOG/ZERO IF NONE
002406                          LBUFN:  .BLKW 1                 ;ADDRESS FOR MORE DATA FOR LOG
002410                          LBUFE:  .BLKW 1                 ;LAST ADDRESS AVAILABLE FOR LOG DATA
                                        
                                        ;DISK DIAGNOSTIC DLL CONTROL WORDS
                                        
002412                          DLL:    .BLKW 1                 ;DOWNLINE LOAD RESPONSE CODE = 0 - NO DATA,
                                                                ;1 - PROGRAM PROVIDED, 2- PROGRAM NOT FOUND
002414                          DLLDR:  .BLKW 1                 ;DRIVE NUMBER REQUESTING PROGRAM
002416                          DLLV:   .BLKW 1                 ;A VALUE FROM DM PROGRAM TO BE RETURNED
002420                          DLLR:   .BLKW 1                 ;REGION
002422                          DLLADR: .BLKW 2                 ;ADDRESS WHERE PROGRAM STORED
002426                          DLLSIZ: .BLKW 1                 ;SIZE OF PROGRAM IN BYTES
002430                          DLLNAM: .BLKW 2                 ;NAME OF PROGRAM IN RAD50
```

```
                                .SBTTL  GLOBAL TEXT SECTION

                                ;++
                                ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
                                ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
                                ; MORE THAN ONE TEST.
                                ;--

                                ;
                                ; NAMES OF DEVICES SUPPORTED BY PROGRAM
                                ;
002434                                  DEVTYP <LOGICAL DISK DRIVE>
002434                                                                  L$DVTYP::
002434  114     117     107                                                     .ASCIZ  /LOGICAL DISK DRIVE/
                                                                                .EVEN

                                ; TEST DESCRIPTION
                                ;
002460                                  DESCRIPT <UDA-50 CONTROLLER AND DISK DRIVE DIAGNOSTIC>
002460                                                                  L$DESC::
002460  125     104     101                                                     .ASCIZ  /UDA-50 CONTROLLER A
                                                                                .EVEN

                                ; DESCRIPTIONS OF INDIVIDUAL TESTS
002534  125     116     111     T1NAME: .ASCIZ\UNIBUS ADDRESSING\
002556  104     111     123     T2NAME: .ASCIZ\DISK RESIDENT\
002574  104     111     123     T3NAME: .ASCIZ\DISK FUNCTION\
002612  104     111     123     T4NAME: .ASCIZ\DISK EXERCISER\
```

G_OBAL TEXT SECTION

```
                                        ;UNFORMATTED MESSAGES USED IN ERROR CALLS
                                        
002631      116     117     124  FMERRM: .ASCIZ\NOT ENOUGH MEMORY. SELECT FEWER UNITS TO TEST.\
002710      127     122     117  RSPPKE: .ASCIZ\WRONG OPCODE RECEIVED IN MESSAGE PACKET\
002760      115     105     123  RSPPRE: .ASCIZ\MESSAGE PACKET RECEIVED WITH WRONG REFERENCE NUMBER\
003044      115     105     123  RSPPNE: .ASCIZ\MESSAGE PACKET RECEIVED WITH UNKNOWN REQUEST NUMBER\
003130      105     122     122  RSPSTE: .ASCIZ\ERROR STATUS CODE REPORTED IN MESSAGE COMMAND\
003206      125     104     101  LOADM1: .ASCIZ\UDA RETURNED ERROR TO SEND STUD DATA COMMAND\
003263      125     104     101  INTHD:  .ASCIZ\UDA INITIALIZE ERROR\
003310      105     122     122  RWRDEM: .ASCIZ\ERROR READING 'DIAGNOSTIC MACHINE'' PROGRAM FILE\
003370      124     127     117  MLDREM: .ASCIZ\TWO P-TABLES POINT TO SAME DRIVE\
003431      115     117     122  TOOMEM: .ASCIZ\MORE THAN EIGHT DRIVES SELECTED ON ONE CONTROLLER\
003513      124     101     102  BADT:   .ASCIZ\TABLE INCONSISTANCY ERROR. START PROGRAM AGAIN.\
003573      116     125     115  T4BB:   .ASCIZ\NUMBER OF BAD BLOCKS\
003620      102     101     104  T4BBI:  .ASCIZ\BAD BLOCK\
003632      122     105     101  T4RO:   .ASCIZ\READ ONLY\
003644      127     122     111  T4WO:   .ASCIZ\WRITE ONLY\
003657      103     110     105  T4WCA:  .ASCIZ\CHECK ALL WRITES BY READING\
003713      122     101     116  T4WCR:  .ASCIZ\RANDOMLY CHECK WRITES BY READING\
003754      104     101     124  T4DP:   .ASCIZ\DATA PATTERN - 0 FOR RANDOM SELECTION\
004022      105     116     101  T4ECC:  .ASCIZ\ENABLE ECC DATA CORRECTION\
004055      103     117     115  T4DCA:  .ASCIZ\COMPARE ALL DATA READ\
004103      122     101     116  T4DCR:  .ASCIZ\RANDOMLY COMPARE DATA READ\
004136      105     116     101  T4RET:  .ASCIZ\ENABLE RETRIES\
004155      122     101     116  T4SEK:  .ASCIZ\RANDOM SEEK MODE\
004176      040     040     000  T4OPT7: .ASCIZ\  \
004201      116     125     115  T4BE:   .ASCIZ\NUMBER OF BEGIN/END SETS\
004232      102     105     107  T4BEG:  .ASCIZ\BEGIN BLOCK\
004246      105     116     104  T4END:  .ASCIZ\END BLOCK\
004260      116     125     115  T4TRC:  .ASCIZ\NUMBER OF TRACKS TO TEST\
004311      124     122     101  T4TRAK: .ASCIZ\TRACK\
004317      116     125     115  T4GRC:  .ASCIZ\NUMBER OF GROUPS TO TEST\
004350      107     122     117  T4GRP:  .ASCIZ\GROUP\
004356      104     117     040  T4CYL:  .ASCIZ\DO YOU WISH TO LIMIT THE CYLINDERS TESTED\
004430      123     124     101  T4CYLB: .ASCIZ\STARTING CYLINDER\
004452      105     116     104  T4CYLE: .ASCIZ\ENDING CYLINDER\
004472      116     125     115  T4DPC:  .ASCIZ\NUMBER OF WORDS IN DATA PATTERN 16\
004535      104     101     124  T4DPD:  .ASCIZ\DATA WORD\
004547      104     115     040  GTDRV1: .ASCIZ\DM PROGRAM ASKED FOR DATA ON UNKNOWN DRIVE\
004622      125     104     101  LOOP00: .ASCIZ\UDASA NON-ZERO AFTER ENTRY INTO LOOP MODE OR WRITING UDASA\
004715      125     104     101  LOOP01: .ASCIZ\UDASA NEVER BECAME NON-ZERO AFTER WRITING WITH NON-ZERO DATA\
005012      104     101     124  LOOP02: .ASCIZ\DATA COMPARISON ERROR DURING DIAGNOSTIC PORT LOOP TEST\
005101      111     116     103  INTST2: .ASCIZ\INCORRECT BR LEVEL\
005124      116     117     040  INTST4: .ASCIZ\NO INTERRUPTS EVER RECEIVED FROM UDA\
005171      101     122     105  INITWC: .ASCIZ\ARE YOU SURE CUSTOMER DATA CAN BE DESTROYED\
005245      106     101     111  DMNF:   .ASCIZ\FAILED TO READ DM PROGRAM FROM DATA FILE\
                                        .EVEN
```

```
                                        ; FORMAT STATEMENTS USED IN PRINT CALLS
                                        
005316  045  124  000  ERRONE: .ASCIZ\%T\
005321  045  116  000  ERRNL:  .ASCIZ\%N\
005324  045  101  124  INTM1:  .ASCIZ\%ATIME-OUT ERROR WHILE WAITING FOR RESPONSE IN UDASA REGISTER%N\
005424  045  101  011  INTM2:  .ASCIZ\%A       UDASA REGISTER = %06%N\
005456  045  101  125  INTM3:  .ASCIZ\%AUDA RESIDENT DIAGNOSTICS DETECTED FAILURE%N\
005534  045  101  125  INTM4:  .ASCIZ\%AUDASA REGISTER DID NOT RETURN CORRECT VALUE%N\
005614  045  101  011  INTM5:  .ASCIZ\%A       EXPECTED = %06%N\
005640  045  101  011  INTM6:  .ASCIZ\%A        ACTUAL  = %06%N\
005664  045  101  122  INTB1M: .ASCIZ\%ARING BUFFER WAS NOT CLEARED BY UDA%N\
005733  045  101  103  INTB2M: .ASCIZ\%ACURRENT CONTENTS OF RING BUFFER:%N\
006000  045  123  066  INTB3M: .ASCIZ\%S6%AADDRESS%S4%ACONTENTS%N\
006034  045  117  061  INTB4M: .ASCIZ\%013%011%N\
006047  045  101  106  RWRDF1: .ASCIZ\%AFILE BEING READ IS '%T%A'%N\
006105  045  101  104  RWRDF2: .ASCIZ\%ADID NOT FIND START AND NULL FRAMES WHERE EXPECTED%N\
006173  045  101  103  RWRDF4: .ASCIZ\%ACHECKSUM ERROR%N\
006216  045  124  045  ERRMB:  .ASCIZ\%T%A  DM PC:%04%A  UDA AT %06%A  \
006260  045  101  104  ERRMBD: .ASCIZ\%ADRIVE %Z3%A  \
006300  045  101  122  ERRMRT: .ASCIZ\%ARUNTIME %D3%A:%Z2%A:%Z2\
006332  045  101  040  ERRME1: .ASCIZ\%A  *  *  *  ERROR PROCESSING DM MESSAGE STRING  *  *  *%N\
006425  045  101  122  MXFERP: .ASCIZ\%AREACHED TRANSFER LIMIT - TESTING STOPPED%N\
006502  045  116  045  ERRLIM: .ASCIZ\%N%AUNIT %Z2%A REACHED ERROR LIMIT - WILL NO LONGER BE TESTED.%N\
006603  045  101  040  LOOP03: .ASCII\%A    DATA SENT     : %07%N\
006636  045  101  040          .ASCIZ\%A    DATA RECEIVED : %07%N\
006672  045  116  045  INTST0: .ASCIZ\%N%ATESTING INTERRUPT ABILITY OF UDA AT ADR %06%A VEC %03%A...\
006771  045  101  103  INTST1: .ASCIZ\%ACOMPLETED%N\
007007  045  101  125  INTST3: .ASCIZ\%AUDA INTERRUPTED AT BR LEVEL %01%N\
007053  045  116  045  INITWA: .ASCIZ\%N%ACUSTOMER DATA WILL BE DESTROYED ON:%N%S5%AUNIT%S5%AUDA AT%S3%ADRIVE%N\
007165  045  123  066  INITWB: .ASCIZ\%S6%D2%S6%06%S4%D3%N\
007212  045  116  045  T4WARN: .ASCIZ\%N%AMANUAL INTERVENTION NOT ALLOWED. TEST 4 USING DEFAULT PARAMETERS%N\
007321  045  116  045  T4QHED: .ASCIZ\%N%ATHE FOLLOWING QUESTIONS REFER TO UNIT %Z2%A UDA AT %06%A DRIVE %Z3%N\
007432  045  116  045  T4OPT1: .ASCIZ\%N%ADO YOU WISH TO:%N\
007460  045  101  040  T4OPT2: .ASCIZ\%A    0 - TEST ENTIRE AREA SELECTED%N\
007525  045  101  040  T4OPT3: .ASCIZ\%A    1 - SPECIFY BEGIN/END SETS TO TEST%N\
007577  045  101  040  T4OPT4: .ASCIZ\%A    2 - SPECIFY TRACKS AND CYLINDERS TO TEST%N\
007657  045  101  040  T4OPT5: .ASCIZ\%A    3 - SPECIFY GROUPS AND CYLINDERS TO TEST%N\
007737  045  101  040  T4OPT6: .ASCIZ\%A    4 - SPECIFY CYLINDERS TO TEST\
010002  045  101  114  INP28A: .ASCIZ\%ALIMITS - LO= 0, HI= 268435455%N\
010044  045  101  111  INP28B: .ASCIZ\%AINVALID CHAR, TYPE DECIMAL NUMBER 0 TO 268435455%N\
010131  045  116  045  MESSG:  .ASCIZ\%N%AUNIT %Z2%A UDA AT %06%A DRIVE %Z3%S\
010201  045  101  104  CTABEM: .ASCIZ\%ADIFFERENT VECTORS, BR LEVELS OR BURST RATES SPECIFIED FOR UDA AT %06%N\
010312  045  116  045  T2WARN: .ASCIZ\%N%AMANUAL INTERVENTION NOT ALLOWED. TEST 2 RUNNING UNATTENDED.%N\
010414  045  116  045  T2CMS1: .ASCIZ\%N%ATEST #2 MANUAL INTERVENTION ON UNIT %Z2%A UDA AT %06%A DRIVE %Z3%N\
010523  045  101  124  T2CMS2: .ASCII\%ATO WRITE AND READ MEMORY:%N\
010560  045  101  040          .ASCII\%A  W DATA REGION OFFSET%N\
010612  045  101  040          .ASCIZ\%A  R REGION OFFSET%N\
010640  045  101  124  T2CMS3: .ASCII\%ATO RUN A DIAGNOSTIC:%N\
010670  045  101  040          .ASCIZ\%A  D REGION%N\
010707  045  101  124  T2CMS4: .ASCII\%ATO EXIT QUESTIONING:%N\
010737  045  101  040          .ASCII\%A  E%N\
010746  045  101  104          .ASCIZ\%ADATA, REGION AND OFFSET ARE HEX VALUES.%N\
011022  045  101  077  T2CMS5: .ASCIZ\%A? INPUT ERROR%N\
011044  045  101  116  NOCLOCK:.ASCIZ\%ANO LINE CLOCK AVAILABLE FOR TIMING EVENTS%N\
011122  045  101  116  RSPTOM: .ASCIZ\%ANO INTERRUPT FROM UDA AT %06%A FOR 3 MINUTES%N\
011203  045  101  106  RSPTMM: .ASCIZ\%AFATAL ERROR FROM UDA AT %06%N%A   UDASA CONTAINS %06%N\
011274  045  101  115  INITMM: .ASCIZ\%AMEMORY ACCESS ERROR TO UDA REGISTERS AT %06%N\
```

GLOBAL TEXT SECTION

```
     75 011354      045      101      125  INTM7:  .ASCIZ\%AUDASA DID NOT GO TO ZERO AFTER STEP 3 WRITE WITH PURGE/POLL BIT SET%N\
     76 011464      040      040      040  INTM8:  .ASCIZ\    UDASA CONTAINS %O6%N\
     77 011514      045      101      105  LOGDAT: .ASCIZ\%AERROR - DATA PLACED IN LOG BUFFER%N\
     78 011562      045      101      105  LOGFUL: .ASCIZ\%AERROR - DATA LOST BECAUSE LOG BUFFER FULL%N\
     79 011640      045      116      045  LOGM1:  .ASCIZ\%N%ACONTENTS OF ERROR LOG:%N\
     80 011675      045      116      045  LOGM2:  .ASCIZ\%N%AEND OF ERROR LOG%N\
     81 011724      045      116      045  LOGM3:  .ASCIZ\%N%AERROR LOG IS EMPTY%N\
     82                                           .EVEN
```

GLOBAL ERROR REPORT SECTION

```
                                .SBTTL  GLOBAL ERROR REPORT SECTION

                                ;++
                                ; THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
                                ; USED BY MORE THAN TEST TO OUTPUT ADDITIONAL ERROR INFORMATION.  PRINTB
                                ; (BASIC) AND PRINTX (EXTENDED) CALLS ARE USED TO CALL PRINT SERVICES.
                                ;--
        177777                  SVCINS= -1                              ; LIST INSTRUCTIONS, SHIFTED RIGHT
        177777                  SVCTST= -1                              ; LIST TEST TAGS, SHIFTED RIGHT
        177777                  SVCSUB= -1                              ; LIST SUBTEST TAGS, SHIFTED RIGHT
        177777                  SVCGBL= -1                              ; LIST GLOBAL TAGS, SHIFTED RIGHT
        177777                  SVCTAG= -1                              ; LIST OTHER TAGS, SHIFTED RIGHT

011756                          BGNMSG INTR1                            ;TIME OUT ERROR
011756                                  PRINTB #INTM1
011776                                  PRINTB #INTM2,R2
012020                          ENDMSG

012022                          BGNMSG INTR2                            ;UDA DIAGS FAILED
012022                                  PRINTB #INTM3
012042                                  PRINTB #INTM2,R2
012064                          ENDMSG

012066                          BGNMSG INTR3                            ;IMPROPER RESPONSE
012066                                  PRINTB #INTM4
012106                                  PRINTB #INTM5,R1
012130                                  PRINTB #INTM6,R2
012152                          ENDMSG

012154                          BGNMSG INTR4                            ;UDA NON-EXISTANT
012154                                  PRINTB #INITMM,(R5)
012176                          ENDMSG

012200                          BGNMSG INTR5
012200                                  PRINTB #INTM7
012220                                  PRINTX #INTM8,R2
012242                          ENDMSG

012244                          BGNMSG INTBF                            ;RING BUFFER NOT CLEARED
012244                                  PRINTB #INTB1M
012264                                  PRINTX #INTB2M
012304                                  PRINTX #INTB3M
012324  013702  002174                  MOV FFREE,R2
012330  010103                          MOV R1,R3
012332                          INTBFL: PRINTX #INTB4M,R2,(R2)
012356  062702  000002                  ADD #2,R2
012362  005303                          DEC R3
012364  003362                          BGT INTBFL
012366                          ENDMSG

012370                          BGNMSG RWRDM1
012370                                  PRINTB #RWRDF1,#FNAME
012414                                  PRINTB #RWRDF2
012434                          ENDMSG

012436                          BGNMSG RWRDM3
012436                                  PRINTB #RWRDF1,#FNAME
```

GLOBAL ERROR REPORT SECTION

```
     58 012462                                  PRINTB #RWRDF4
     59 012502                        ENDMSG
     60
     61 012504                        BGNMSG  LOOP
     62 012504                                PRINTB  #LOOP03,R2,2(R4)
     63 012532                        ENDMSG
     64
     65 012534                        BGNMSG  LOOPA
     66 012534                                PRINTB  #LOOP03,#140000,2(R4)
     67 012564                        ENDMSG
     68
     69 012566                        BGNMSG  INTERR
     70 012566                                PRINTB  #INTST3,R1
     71 012610                        ENDMSG
     72
     73 012612                        BGNMSG CTABE
     74 012612                                PRINTB #CTABEM,(R3)
     75 012634                        ENDMSG
     76
     77 012636                        BGNMSG RSPTOE
     78 012636                                PRINTB #RSPTOM,(R5)
     79 012660                        ENDMSG
     80
     81 012662                        BGNMSG RSPTME
     82 012662                                PRINTB #RSPTMM,(R5),R1
     83 012706                        ENDMSG
     84
     85          000001               SVCINS= 1                  ; LIST INSTRUCTIONS, SHIFTED RIGHT
     86          000001               SVCTST= 1                  ; LIST TEST TAGS, SHIFTED RIGHT
     87          000001               SVCSUB= 1                  ; LIST SUBTEST TAGS, SHIFTED RIGHT
     88          000001               SVCGBL= 1                  ; LIST GLOBAL TAGS, SHIFTED RIGHT
     89          000001               SVCTAG= 1                  ; LIST OTHER TAGS, SHIFTED RIGHT
```

```
      1                                 .SBTTL  GLOBAL SUBROUTINES SECTION
      2
      3                                 ;MEMORY ALLOCATION ERROR
      4                                 ;
      5                                 ;THIS ROUTINE PRINTS A SYSTEM FATAL ERROR AND EXITS THE TEST
      6
      7 012710                          FMERR:  ERRSF 1,FMERRM
        012710  104454                                                          TRAP    C$ERSF
        012712  000001                                                          .WORD   1
        012714  002631                                                          .WORD   FMERRM
        012716  000000                                                          .WORD   0
      8 012720                                  DOCLN                         ;ABORT
        012720  104444                                                          TRAP    C$DCLN
```

GLOBAL SUBROUTINES SECTION

```
                                        ;ALOCM
                                        ;
                                        ;ALLOCATE A BLOCK OF FREE MEMORY.  REPORT ERROR IF MEMORY EXHAUSTED.
                                        ;
                                        ;INPUTS:
                                        ;       R1 - NUMBER OF WORDS TO ALLOCATE
                                        ;       FFREE - FIRST FREE WORD IN MEMORY
                                        ;       FSIZE - SIZE OF FREE MEMORY AVAILABLE IN WORDS
                                        ;OUTPUTS:
                                        ;       R1 - ADDRESS OF FIRST WORD OF ALLOCATED MEMORY
                                        ;       FFREE - NEW FIRST FREE WORD IN MEMORY
                                        ;       FSIZE - SIZE OF FREE MEMORY LEFT AFTER ALLOCATION
                                        ;SYSTEM FATAL ERROR WILL BE REPORTED IF NOT ENOUGH MEMORY AVAILABLE
                                        ;AND ENTIRE PROGRAM WILL BE STOPPED.

016 012722                              ALOCM:  PUSH FFREE                      ;SAVE FFREE AT ENTRY
    012722  013746  002174                              MOV FFREE,-(SP)
017 012726  160137  002176                      SUB R1,FSIZE                    ;REDUCE SIZE OF FREE MEMORY
018 012732  002766                              BLT FMERR                       ;REPORT ERROR IF NOT ENOUGH MEMORY
019 012734  060101                              ADD R1,R1                       ;CHANGE WORDS TO BYTES
020 012736  060137  002174                      ADD R1,FFREE                    ;CALCULATE NEW START OF FREE MEMORY
021 012742                                      POP R1                          ;GET START OF ALLOCATED MEMORY
    012742  012601                                      MOV (SP)+,R1
022 012744  000207                              RETURN
```

```
      1                                 ;HCOMM
      2                                 ;
      3                                 ;ALLOCATES MEMORY FOR HOST COMM AREA AND PACKET BUFFERS WITH ONE
      4                                 ;DESCRIPTOR IN EACH RING. TO BE CALLED AFTER INITIALIZING
      5                                 ;A CONTROLLER WITH SA.MSG=0 AND SA.CMD=0.
      6                                 ;
      7                                 ;INPUTS:
      8                                 ;       R5 - ADDRESS OF CONTROLLER TABLE
      9                                 ;OUTPUTS:
     10                                 ;       CONTROLLER TABLE POINTING TO HOST COMM AREA
     11                                 ;       RING POINTERS TO PACKETS
     12                                 ;       R4 - ADDRESS OF HOST COMM AREA
     13
     14 012746  012701  000166          HCOMM:  MOV #HC.SIZ/2,R1                ;GET SIZE OF AREA TO ALLOCATE
     15 012752  004737  012722                  CALL ALOCM                      ;ALLOCATE THE MEMORY
     16 012756  010104                          MOV R1,R4                       ;GET ADDRESS OF HOST COMM AREA
     17 012760  010465  000016                  MOV R4,C.RING(R5)               ;PLACE IN CONTROLLER TABLE
     18 012764  062701  000020                  ADD #HC.MPK,R1                  ;COMPUTE START OF MESSAGE PACKET
     19 012770  010164  000004                  MOV R1,HC.MSG(R4)               ;PLACE IN RING
     20 012774  062701  000064                  ADD #<HC.CPK-HC.MPK>,R1         ;COMPUTE START OF COMMAND PACKET
     21 013000  010164  000010                  MOV R1,HC.CMD(R4)               ;PLACE IN RING
     22 013004  000207                          RETURN
```

```
                                        ;TINIT
                                        ;
                                        ;INITIALIZE VARIABLES FOR TEST
                                        ;
                                        ;INPUTS:
                                        ;       R1 - TEST NUMBER
                                        ;OUTPUTS:
                                        ;       LBUFS - CLEARED (DELETES ERROR LOG)
                                        ;       FFREE - FROM FMEM
                                        ;       FSIZE - FROM FMEMS
                                        ;       TNUM - TEST NUMBER FROM R1
                                        ;       ALL REGISTERS CLOBERED
                                        ;
013006  004737  024414          TINIT:  CALL RESET                      ;RESET ALL DEVICES
013012  005037  002404                  CLR LBUFS                       ;CLEAR ERROR LOG BUFFER POINTER
013016  013737  002200  002174          MOV FMEM,FFREE                  ;INIT FFREE
013024  013737  002202  002176          MOV FMEMS,FSIZE                 ;INIT FSIZE
013032  020137  002226                  CMP R1,TNUM                     ;SEE IF SAME TEST RUNNING
013036  001007                          BNE TINITR                      ;IF NOT, GO TO READ DM PROGRAM
013040  013737  002214  002174          MOV DMEND,FFREE                 ;CHANGE FREE MEMORY TO LEAVE
013046  013737  002216  002176          MOV DMENDS,FSIZE                ;  DM PROGRAM ALLOCATED
013054  000207                          RETURN
                                        ;
013056                          TINITR:                                 ;BRESET ;RESET ALL UNITS
013056  004737  013070                  CALL READDM                     ;READ DM PROGRAM
013062  010137  002226                  MOV R1,TNUM                     ;STORE TEST NUMBER TO SHOW DM PROGRAM IN MEMORY
013066  000207                          RETURN
```

```
                                        ;READDM
                                        ;
                                        ;READ A DM PROGRAM INTO FREE MEMORY
                                        ;
                                        ;INPUTS:
                                        ;       R1 - TEST NUMBER
                                        ;OUTPUTS:
                                        ;       DMPROG - POINTER TO START OF DM PROGRAM IN MEMORY
                                        ;       R1 - UNCHANGED
                                        ;       CARRY CLEAR IF NO ERROR, CARRY SET IF PROGRAM NOT FOUND
                                        ;ALL REGISTERS BUT R1 ARE USED AND PREVIOUS CONTENTS DESTROYED
                                        
013070  013737  002174  002212  READDM: MOV FFREE,DMPROG                ;GET STORAGE ADDRESS
013076  004737  023734                  CALL RDREC
013102  103407                          BCS README                      ;CHECK IF ERROR
013104  013737  002174  002214          MOV FFREE,DMEND                 ;SAVE END OF ADDRESS OF DM PROGRAM
013112  013737  002176  002216          MOV FSIZE,DMENDS                ;  AND CURRENT SIZE OF FREE MEMORY
013120  000207                          RETURN

013122                          README: ERRSF 2,DMNF                    ;REPORT DM PROGRAM NOT FOUND
013122  104454                                                                  TRAP    C$ERSF
013124  000002                                                                  .WORD   2
013126  005245                                                                  .WORD   DMNF
013130  000000                                                                  .WORD   0
013132                                  DOCLN
013132  104444                                                                  TRAP    C$DCLN
```

```
                                        ;RUNDM
                                        ;
                                        ;LOAD AND RUN A DM PROGRAM IN THE CONTROLLERS. RETURN WHEN ALL
                                        ;DM PROGRAMS HAVE TERMINATED.
                                        ;
                                        ;INPUTS:
                                        ;       TSTTAB - POINTER TO FIRST CONTROLLER TABLE
                                        ;       R1 - NUMBER OF CONTROLLERS TO TEST
                                        ;IMPLICIT INPUTS:
                                        ;       DMPROG - POINTER TO START OF DM PROGRAM IN MEMORY
                                        ;OUTPUTS:
                                        ;       Z SET IF NO CONTROLLERS SUCCESSFULLY STARTED
                                        ;ALL REGISTERS ARE USED AND PREVIOUS CONTENTS DESTROYED.

013134  010137  002230          RUNDM:  MOV R1,URUN                     ;SAVE NUMBER OF UNITS TO RUN
013140  005037  002232                  CLR URNING                      ;CLEAR NUMBER OF UNITS RUNNING

                                        ;LOAD DM PROGRAM INTO EACH CONTROLLER

013144  013737  002230  002234          MOV URUN,UCNT                   ;SET COUNTER OF UNITS
013152  013705  002210                  MOV TSTTAB,R5                   ;GET FIRST CONTROLLER TABLE
013156  005065  000014          LDDM:   CLR C.FLG(R5)                   ;CLEAR ALL FLAGS
013162  016537  000002  002074          MOV C.UNIT(R5),L$LUN            ;SEE IF UNIT TO BE TESTED
013170  100405                          BMI LDNEXT                      ;IF NOT, DON'T LOAD THIS UNIT
013172                                          ASSUME CT.AVL EQ BIT15
013172  004737  021204                  CALL LOADDM                     ;LOAD THE DM PROGRAM
013176  001402                          BEQ LDNEXT                      ;IF ERROR, GO TO NEXT CONTROLLER
013200  005237  002232                  INC URNING                      ;IF NO ERROR, COUNT UNIT RUNNING
013204  062705  000046          LDNEXT: ADD #C.SIZE,R5                  ;MOVE TO NEXT CONTROLLER TABLE
013210  005337  002234                  DEC UCNT                        ;CHECK IF MORE CONTROLLERS
013214  001360                          BNE LDDM                        ;LOAD NEXT

                                        ;CHECK IF ANY CONTROLLERS LOADED

013216  005737  002232                  TST URNING                      ;ANY UNITS LOADED?

                                        ;THE DM PROGRAMS ARE NOW IN CONTROL
                                        ;RESPDM MUST BE CALLED TO RESPOND TO THEIR REQUESTS

013222  000207                          RETURN
```

```
                                        ;RESPDM
                                        ;
                                        ;RESPOND TO DM REQUESTS. RETURN WHEN ALL DM PROGRAMS
                                        ;HAVE TERMINATED.

013224  013705  002210                  RESPDM: MOV TSTTAP,R5                   ;GET CONTROLLER TABLE ADDRESS
013230  013737  002230  002234                  MOV URUN,UCNT                   ;SET COUNTER OF UNITS
013236  016504  000016                  RESPCT: MOV C.RING(R5),R4               ;GET HOST COMM AREA ADDRESS
013242  032765  000002  000014                  BIT #CT.RN,C.FLG(R5)            ;CHECK IF PROGRAM RUNNING
013250  001446                                  BEQ RSPNXT                      ;IF NOT, LOOK AT NEXT
013252  016537  000002  002074                  MOV C.UNIT(R5),L$LUN            ;STORE UNIT NUMBER UNDER TEST
013260  032765  000010  000014                  BIT #CT.MSG,C.FLG(R5)           ;SEE IF INTERRUPT RECEIVED
013266  001077                                  BNE RSPIN                       ;IF SO, LOOK AT PACKET
013270  032765  000004  000014                  BIT #CT.CMD,C.FLG(R5)           ;SEE IF COMMAND HAS BEEN SENT
013276  001540                                  BEQ RSPOU                       ;IF NOT, SEND ONE

                                        ;CHECK IF UDA STILL RUNNING

013300  011503                                  MOV (R5),R3                     ;GET ADDRESS OF UDAIP
013302  016301  000002                          MOV 2(R3),R1                    ;LOOK AT UDASA REGISTER
013306  001405                                  BEQ RSPTM                       ;IF ZERO, UDA STILL RUNNING
013310                                          ERRDF 35,0,RSPTME               ;REPORT UDA HAS FATAL ERROR
013310  104455                                                                          TRAP    C$ERDF
013312  000043                                                                          .WORD   35
013314  000000                                                                          .WORD   0
013316  012662                                                                          .WORD   RSPTME
013320  000453                                  BR RSPDRP                       ;DROP CONTROLLER FROM TESTING

                                        ;CHECK FOR TIMEOUT OF RESPONSE

013322                                  RSPTM:
013322  005737  002352                          TST KW.CSR                      ;SEE IF A CLOCK ON SYSTEM
013326  001416                                  BEQ RSPNTO                      ;DON'T TIME IF NO CLOCK
013330  023765  002364  000042                  CMP KW.EL+2,C.TOH(R5)           ;COMPARE TO TIMEOUT COUNTER
013336  101005                                  BHI RSPTMO
013340  001011                                  BNE RSPNTO
013342  023765  002362  000040                  CMP KW.EL,C.TO(R5)
013350  103405                                  BLO RSPNTO                      ;IF TOO MUCH TIME ELAPSED SINCE LAST INTERRUPT
013352                                  RSPTMO: ERRDF 36,0,RSPTOE               ;REPORT TIMEOUT ERROR
013352  104455                                                                          TRAP    C$ERDF
013354  000044                                                                          .WORD   36
013356  000000                                                                          .WORD   0
013360  012636                                                                          .WORD   RSPTOE
013362  000432                                  BR RSPDRP                       ;DROP CONTROLLER FROM TESTING
013364                                  RSPNTO:
013364                                          BREAK                           ;ALLOW DRS TO SEE TERMINAL INPUT
013364  104422                                                                          TRAP    C$BRK

                                        ;CHECK FOR TIME TO PRINT STATISTICAL REPORT

013366  005737  002352                  RSPNXT: TST KW.CSR                      ;ANY CLOCK ON SYSTEM?
013372  001420                                  BEQ RSPNRP                      ;BYPASS IF NOT
013374  023737  002364  002374                  CMP KW.EL+2,STIME+2             ;  A STATISTICAL REPORT
013402  101005                                  BHI RSPRPT
013404  001013                                  BNE RSPNRP
013406  023737  002362  002372                  CMP KW.EL,STIME
013414  103407                                  BLO RSPNRP
```

```
                                RSPRPT: DORPT                           ;PRINT THE REPORT
013416  104424                                                                  TRAP    C$DRPT
013420  012700  001604                  MOV #15.*60.,R0                 ;SET TIME FOR NEXT REPOR
013424  012701  002372                  MOV #STIME,R1                   ;AT 15 MINUTES FROM NOW
013430  004737  022660                  CALL SETTO

                                ;SWITCH TO NEXT CONTROLLER

013434  062705  000046          RSPNRP: ADD #C.SIZE,R5                  ;MOVE TO NEXT TABLE
013440  005337  002234                  DEC UCNT                        ;CHECK IF MORE CONTROLLERS
013444  001274                          BNE RESPCT                      ;LOOK AT NEXT CONTROLLER
013446  000666                          BR RESPDM                       ;LOOK AT FIRST CONTROLLER AGAIN

                                ;REMOVE A CONTROLLER FROM TESTING

013450  042765  000012  000014  RSPDRP: BIC #CT.RN+CT.MSG,C.FLG(R5)     ;CLEAR PROGRAM RUNNING
013456  005337  002232                  DEC URNING                      ;REDUCE RUNNING CONTROLLERS COUNT
013462  001341                          BNE RSPNXT                      ;IF ANY STILL RUNNING, LOOK AT THEM
013464  000207                          RETURN                          ;ELSE RETURN TO TEST SECTION
```

```
                                        ;CONTROLLER HAS RESPONDED, LOOK AT MESSAGE PACKET
                                        ;
                                        ;CHECK FOR PROPER OPCODE IN END PACKET
                                        ;
013466  012700  000204          RSPIN:  MOV #OP.END+OP.SSD,R0           ;GET SEND DATA END PACKET OPCODE
013472  032765  000020  000014          BIT #CT.REQ,C.FLG(R5)           ;LOOK IF SEND DATA OR RECEIVE DATA
013500  001402                          BEQ RSPMWR
013502  012700  000205                  MOV #OP.END+OP.RSD,R0           ;CHANGE TO RECEIVE DATA END PACKET OPCODE
013506  120064  000030          RSPMWR: CMPB R0,HC.MPK+P.OPCD(R4)       ;COMPARE TO OPCODE IN END PACKET
013512  001405                          BEQ RSPSTS
013514                                  ERRHRD 25,RSPPKE                ;REPORT OPCODE ERROR IN RESPONSE PACKET
013514  104456                                                                  TRAP    C$ERHRD
013516  000031                                                                  .WORD   25
013520  002710                                                                  .WORD   RSPPKE
013522  000000                                                                  .WORD   0
013524  000751                          BR RSPDRP                       ;DROP UNIT FROM TESTING
                                        ;
                                        ;LOOK AT STATUS CODE
                                        ;
013526  032764  000037  000032  RSPSTS: BIT #ST.MSK,HC.MPK+P.STS(R4)    ;CHECK FOR STATUS CODE ST.SUC (ZERO)
013534  001405                          BEQ RSPREF
013536                                  ERRHRD 26,RSPSTE                ;REPORT ERROR STATUS CODE
013536  104456                                                                  TRAP    C$ERHRD
013540  000032                                                                  .WORD   26
013542  003130                                                                  .WORD   RSPSTE
013544  000000                                                                  .WORD   0
013546  000740                          BR RSPDRP                       ;DROP UNIT FROM TESTING
                                        ;
                                        ;CHECK FOR EXPECTED REFERENCE NUMBER
                                        ;
013550  026564  000044  000020  RSPREF: CMP C.REF(R5),HC.MPK+P.CRF(R4)  ;CHECK IF CORRECT REF NUMBER
013556  001405                          BEQ RSPPTW
013560                                  ERRHRD 31,RSPPRE
013560  104456                                                                  TRAP    C$ERHRD
013562  000037                                                                  .WORD   31
013564  002760                                                                  .WORD   RSPPRE
013566  000000                                                                  .WORD   0
013570  000727                          BR RSPDRP                       ;DROP UNIT FROM TESTING
                                        ;
                                        ;CHECK IF RESPONSE FROM SEND OR RECEIVE DATA COMMAND
                                        ;
013572  032765  000020  000014  RSPPTW: BIT #CT.REQ,C.FLG(R5)           ;CHECK IF RESPONSE FROM DM PROGRAM
013600  001433          RSPOU:  BEQ RSPOUT                      ;LOOK AT REQUEST NUMBER IF SO
```

GLOBAL SUBROUTINES SECTION

```
                                        ;MAINTENANCE READ END PACKET RECEIVED, LOOK AT REQUEST FROM DM PROGRAM

013602  016401  000260        RSPPT2: MOV HC.BF2(R4),R1               ;GET REQUEST NUMBER
013606  020127  000017                CMP R1,#DSPSIZ                  ;CHECK IF IN EXPECTED RANGE
013612  103405                        BLO RSPPT3
013614                                ERRHRD 32,RSPPNE                ;BAD REQUEST NUMBER
                                                                              TRAP    C$ERHRD
013614  104456                                                          .WORD   32
013616  000040                                                        .WORD   RSPPNE
013620  003044                                                        .WORD   0
013622  000000
013624  000711                        BR RSPDRP                       ;DROP UNIT FROM TESTING

013626  012700  000004        RSPPT3: MOV #OP.SSD,R0                  ;BUILD A SEND DATA COMMAND PACKET
013632  004737  021624                CALL BLDCMD                     ;  FOR ANSWER TO DM PROGRAM
013636  012700  000164                MOV #HC.BF1,R0                  ;POINT TO BUFFER IN PACKET
013642  004737  022006                CALL CLRBUF                     ; AND CLEAR BUFFER
013646  010403                        MOV R4,R3                       ;R3 POINTS TO COMMAND BUFFER
013650  062704  000074                ADD #HC.BSZ,R4                  ;R4 POINTS TO MESSAGE BUFFER
013654  011401                        MOV (R4),R1                     ;GET REQUEST NUMBER
013656  012423                        MOV (R4)+,(R3)+                 ;PUT REQUEST NUMBER INTO COMMAND PACKET
013660  060101                        ADD R1,R1                       ;DOUBLE REQUEST NUMBER
013662  004771  013772                CALL @RSPDSP(R1)                ;CALL REQUESTED ROUTINE
013666  001270                        BNE RSPDRP                      ;ROUTINE RETURNS Z CLEAR TO DROP UNIT FROM TESTING
                                                                      ; Z SET IF COMMAND READY TO SEND TO UNIT


                                        ;SEND COMMAND BACK TO UDA

013670  042765  000010  000014  RSPOUT: BIC #CT.MSG,C.FLG(R5)           ;CLEAR MESSAGE RECEIVED FLAG
013676  032765  000020  000014          BIT #CT.REQ,C.FLG(R5)           ;CHECK WHICH COMMAND TO SEND
013704  001014                        BNE RSPOU2                      ;BRANCH IF RESPONSE TO REQUEST

013706  012700  000005                MOV #GP.RSD,R0                  ;BUILD RECEIVE DATA COMMAND
013712  004737  021624                CALL BLDCMD
013716  012700  000260                MOV #HC.BF2,R0                  ;POINT TO MESSAGE BUFFER
013722  004737  022006                CALL CLRBUF                     ;  AND CLEAR IT
013726  052765  000020  000014        BIS #CT.REQ,C.FLG(R5)           ;SET REQUEST BIT
013734  000403                        BR RSPOU3

013736  042765  000020  000014  RSPOU2: BIC #CT.REQ,C.FLG(R5)         ;CLEAR REQUEST BIT
013744                          RSPOU3:
013744  004737  021710                CALL SNDCMD                     ;SEND COMMAND TO UDA
013750  012700  000264                MOV #3.*60.,R0                  ;SET TIMEOUT FOR 3 MINUTES
013754  010501                        MOV R5,R1
013756  062701  000040                ADD #C.TO,R1                    ;PUT TIME IN CONTROLLER TABLE
013762  004737  022660                CALL SETTO
013766  000137  013366                JMP RSPNXT                      ;NOW WAIT FOR END PACKET


                                        ;RESPONSE REQUEST DISPATCH TABLE

013772  014030          RSPDSP: .WORD T1MSIZ                          ; 0. SET UP FREE MEMORY FOR ADDRESS TESTING
013774  014150                  .WORD T2DLL                           ; 1. PROVIDE DIAGNOSTIC PROGRAM FOR DISK DRIVE
013776  014314                  .WORD T2CMD                           ; 2. GET MANUAL INTERVENTION COMMAND
014000  015150                  .WORD T4MPRM                          ; 3. TELL DATA PATTERN 16.
014002  015172                  .WORD T4UPRM                          ; 4. TELL UNIT PARAMETERS, CLEAR CONTENTS
014004  015520                  .WORD T4BB1                           ; 5. TELL BAD BLOCKS (FIRST 14)
014006  015550                  .WORD T4BB2                           ; 6. TELL BAD BLOCKS (LAST TWO)
014010  015600                  .WORD T4SOFT                          ; 7. ADD TO SOFT ERROR AND ECC COUNTS
```

```
     54 014012  015626                    .WORD T4SEEK                ; 8. ADD 1000 TO SEEK COUNT
     55 014014  015646                    .WORD T4MXFR                ; 9. ADD TO MEGABITS READ AND WRITE COUNTS
     56 014016  016010                    .WORD UTOTST                ;10. TELL WHICH DRIVES TO TEST
     57 014020  016076                    .WORD ERRMES                ;11. REPORT ERROR MESSAGE
     58 014022  016344                    .WORD ERRMC                 ;12. REPORT ERROR MESSAGE AND COUNT HARD ERROR
     59 014024  016466                    .WORD MESSAG                ;13. PRINT A DESCRIPTIVE MESSAGE
     60 014026  016576                    .WORD DONE                  ;14. MARK DM PROGRAM AS NO LONGER RUNNING
     61
     62         000017                    DSPSIZ=<.-RSPDSP>/2         ;LEGAL NUMBERS ARE LOWER THAN THIS
```

```
                        ;NORMAL MAINTENANCE READ BUFFER DESCRIPTION
                        ;
                        ;BYTE OFFSET FROM
                        ;START OF BUFFER
                        ;       0       +-------------------------------+
                        ;               !       REQUEST NUMBER          !  USED TO SELECT ROUTINE
                        ;               +-------------------------------+
                        ;       2       !       DATA ARGUMENT #1        !  R4 CONTAINS THIS ADDRESS
                        ;               +-------------------------------+
                        ;       4       !       DATA ARGUMENT #2        .
                        ;               +-------------------------------+
                        ;       6       !       DATA ARGUMENT #3        .
                        ;               +-------------------------------+
                        ;       8       !       DATA ARGUMENT #4        .
                        ;               +-------------------------------+
                        ;       10      !       DATA ARGUMENT #5        .
                        ;               +-------------------------------+
                        ;       12      !       DATA ARGUMENT #6        .
                        ;               +-------------------------------+
                        ;       14      !       DATA ARGUMENT #7        .
                        ;               +-------------------------------+
                        ;       16      !       DATA ARGUMENT #8        .
                        ;               +-------------------------------+
                        ;       18      !       DATA ARGUMENT #9
                        ;               +-------------------------------+
                        ;       20      !       DATA ARGUMENT #10       .
                        ;               +-------------------------------+
                        ;       22      .       DATA ARGUMENT #11       .
                        ;               +-------------------------------+
```

```
;NORMAL PSEUDO-TERMINAL IN PACKET DESCRIPTION GIVEN IN RESPONSE TO ABOVE PACKET

;BYTE OFFSET FROM
;START OF PACKET
;        0        +-------------------------------+
;                 !        REQUEST NUMBER         !  ECHOED FROM REQUEST PACKET
;        2        +-------------------------------+
;                 !        DATA ARGUMENT #1       !  R3 CONTAINS THIS ADDRESS
;        4        +-------------------------------+
;                 !        DATA ARGUMENT #2       !  ALL DATA ARGUMENTS ARE RETURNED
;        6        +-------------------------------+  CONTAINING ZEROS UNLESS
;                 !        DATA ARGUMENT #3       !  SPECIFICALLY INDICATED BY
;        8        +-------------------------------+  RESPONSE ROUTINE.
;                 !        DATA ARGUMENT #4       .
;       10        +-------------------------------+
;                 !        DATA ARGUMENT #5       .
;       12        +-------------------------------+
;                 !        DATA ARGUMENT #6       .
;       14        +-------------------------------+
;                 !        DATA ARGUMENT #7       !
;       16        +-------------------------------+
;                 !        DATA ARGUMENT #8       .
;       18        +-------------------------------+
;                 !        DATA ARGUMENT #9       .
;       20        +-------------------------------+
;                 !        DATA ARGUMENT #10
;       22        +-------------------------------+
;                 .        DATA ARGUMENT #11
;                 +-------------------------------+
```

```
                                        ;T1MSIZ  -  DM REQUEST 0
                                        ;
                                        ;SET UP MEMORY FOR ADDRESS TESTING FROM UDA.
                                        ;PLACE ADDRESS OF EACH LOCATION INTO EACH LOCATION IN FREE
                                        ;MFMORY. RETURN FIRST LOCATION OF FREE MEMORY IN CMD.02 (LOW BITS)
                                        ;AND CMD.03 (HIGH BITS). RETURN LAST LOCATION OF FREE MEMORY IN
                                        ;CMD.04 AND CMD.05. ALSO RETURN FIRST EXISTANT LOCATION IN CMD.06
                                        ;AND CMD.07; LAST EXISTANT LOCATION IN CMD.08 AND CMD.09.
                                        ;
                                        ;INPUTS:
                                        ;       R5 - CONTROLLER TABLE ADDRESS
                                        ;       R4 - MESSAGE PACKET DATA ADDRESS (POINTING TO MSG.02)
                                        ;       R3 - COMMAND PACKET DATA ADDRESS (POINTING TO CMD.02)
                                        ;OUTPUTS:
                                        ;       COMMAND PACKET CONTAINING:
                                        ;         (R3) LOW ADDRESS BITS OF FIRST WRITABLE ADDRESS
                                        ;       2.(R3) HIGH ADDRESS BITS OF FIRST WRITABLE ADDRFSS
                                        ;       4.(R3) LOW ADDRESS BITS OF LAST WRITABLE ADDRESS
                                        ;       6.(R3) HIGH ADDRESS BITS OF LAST WRITABLE ADDRESS
                                        ;       8.(R3) LOW ADDRESS BITS OF FIRST READABLE ADDRESS
                                        ;      10.(R3) HIGH ADDRESS BITS OF FIRST READABLE ADDRESS
                                        ;      12.(R3) LOW ADDRESS BITS OF LAST READABLE ADDRESS
                                        ;      14.(R3) HIGH ADDRESS BITS OF LAST READABLE ADDRESS
                                        ;       Z SET
                                        ;
014030  013701  002174          T1MSIZ: MOV FFREE,R1                    ;GET FIRST ADDRESS OF FREE MEMORY
014034  013702  002176                  MOV FSIZE,R2                    ;GET SIZE
                                        ;
                                        ;FILL MEMORY WITH ADDRESS PATTERN
                                        ;
014040  010111                  MEMFIL: MOV R1,(R1)                     ;WRITE DATA INTO LOCATION
014042  062701  000002                  ADD #2,R1                       ;INCREASE ADDRESS TO NEXT LOCATION
014046  005302                          DEC R2                          ;COUNT THE WORDS
014050  001373                          BNE MEMFIL                      ;FILL ALL WORDS
                                        ;
                                        ;SEND LOCATION OF FREE MEMORY TO UDA
                                        ;
014052  013723  002174                  MOV FFREE,(R3)+                 ;LOAD FIRST ADDRESS OF FREE MEMORY
014056  005023                          CLR (R3)+                       ; HIGH ORDER BITS ARE ZERO
014060  013700  002176                  MOV FSIZE,R0                    ;GET SIZE OF FREE MEMORY
014064  006300                          ASL R0                          ;CONVERT TO BYTES
014066  063700  002174                  ADD FFREE,R0                    ;COMPUTE LAST LOCATION
014072  162700  000002                  SUB #2,R0
014076  010023                          MOV R0,(R3)+                    ;LOAD LAST LOCATION
014100  005023                          CLR (R3)+                       ;CLEAR HIGH ORDER BITS
                                        ;
                                        ;SEND LOCATION OF READABLE MEMORY
                                        ;
014102  005023                          CLR (R3)+                       ;SEND ZERO AS START OF READABLE MEMORY
014104  005023                          CLR (R3)+
014106  013700  002120                  MOV L$HIMEM,R0                  ;GET HIGH MEMORY ADDRESS
014112  005001                          CLR R1                          ;CLEAR HIGH BITS
014114  006300                          ASL R0                          ;SHIFT LEFT 6 PLACES
014116  006300                          ASL R0
014120  006300                          ASL R0
014122  006300                          ASL R0
014124  006300                          ASL R0
```

```
     58 014126  006101                    ROL R1
     59 014130  006300                    ASL R0
     60 014132  006101                    ROL R1
     61 014134  052700  000076            BIS #76,R0          ;SET LOW ORDER BITS
     62 014140  010023                    MOV R0,(R3)+        ;PUT INTO BUFFER
     63 014142  010123                    MOV R1,(R3)+
     64 014144  000264                    SEZ
     65 014146  000207                    RETURN
```

GLOBAL SUBROUTINES SECTION

```
                        ;T2DLL - DM REQUEST 1
                        ;
                        ;PROVIDE DIAGNOSTIC TO DOWNLINE LOAD INTO DISK DRIVE.
                        ;
                        ;THE UDA MAY BE USED TO GET THE DIAGNOSTIC IF THE SYSTEM LOAD DEVICE
                        ;IS ON THE UDA. THIS ACTION WILL CAUSE A REINITIALIZATION OF THE UDA
                        ;AND THE RING STRUCTURE MOVED. SINCE THIS PROGRAM HAS NO WAY TO
                        ;DETERMINE IF THE UDA IS USED, IT WILL ALWAYS ASSUME IT IS USED AND
                        ;WILL INITIALIZE AND RELOAD THE DM PROGRAM AFTER THEREADING THE
                        ;DIAGNOSTIC. THE OUTPUTS OF THIS ROUTINE ARE STORED AND SENT TO THE
                        ;DM PROGRAM IN THE UTOTST REQUEST.
                        ;
                        ;INPUTS:
                        ;       R5 - CONTROLLER TABLE ADDRESS
                        ;       R4 - MESSAGE DATA ADDRESS
                        ;          (R4) DRIVE NUMBER
                        ;        2.(R4) A VALUE THE DM PROGRAM WISHES RETURNED
                        ;        4.(R4) REGION TO WHICH PROGRAM IS TO BE LOADED IN DISK
                        ;        6.(R4) 2 WORD PROGRAM NAME IN RAD50
                        ;       R3 - COMMAND DATA ADDRESS
                        ;OUTPUTS:
                        ;       COMMAND PACKET COULD CONTAIN THE FOLLOWING:
                        ;          (R3) ONE IF PROGRAM PROVIDED, TWO IF PROGRAM NOT AVAILABLE
                        ;        2.(R3) DRIVE NUMBER
                        ;        4.(R3) COPY OF THE VALUE FROM DM PROGRAM
                        ;        6.(R3) REGION TO WHICH PROGRAM IS TO BE LOADED
                        ;        8.(R3) ADDRESS OF FIRST BYTE TO BE DOWNLINE LOADED
                        ;       10.(R3) HIGH ORDER BITS OF ADDRESS
                        ;       12.(R3) BYTE COUNT OF PROGRAM TO BE DOWNLINE LOADED
                        ;       Z SET
                        ;THIS PROGRAM WILL NOT SEND A COMMAND PACKET IN RESPONSE TO THIS REQUEST.
                        ;THE UDA WILL BE REINITIALIZED AND THE DM PROGRAM RELOADED. THEN THIS DATA
                        ;WILL BE APPENDED TO THE NEXT UTOTST REQUEST.

                        ;COPY REQUEST DATA TO STORAGE

014150  005037  002412  T2DLL:  CLR DLL                  ;CLEAR CONTROL WORD
014154  012437  002414          MOV (R4)+,DLLDR         ;DRIVE NUMBER
014160  012437  002416          MOV (R4)+,DLLV          ;VALUE FROM DM
014164  012437  002420          MOV (R4)+,DLLR          ;REGION
01417J  012437  002430          MOV (R4)+,DLLNAM        ;PROGRAM NAME
014174  012437  002432          MOV (R4)+,DLLNAM+2      ;  (TWO WORDS)

                        ;RESET UDA AND READ DM PROGRAM

014200  005075  000000          CLR @(R5)               ;RESET THE UDA
014204  013737  002174  002422  MOV FFREE,DLLADR        ;GET ADDRESS WHERE PROGRAM
014212  005037  002424          CLR DLLADR+2            ;  TO BE STORED
014216  013737  002176  002426  MOV FSIZE,DLLSIZ        ;SAVE CURRENT SIZE OF MEMORY
014224  004737  023674          CALL RDDLL              ;READ DLL PROGRAM FROM DATA FILE
014230  103002                  BCC 1$                  ;PROGRAM NOT FOUND IF CARRY SET
014232  005237  002412          INC DLL                 ;RETURN 1 IF PROGRAM FOUND
014236  005237  002412  1$:     INC DLL                 ;RETURN 2 IF PROGRAM NOT FOUND
014242  013737  002426  002176  MOV DLLSIZ,FSIZE        ;COMPUTE SIZE OF DLL PROGRAM
014250  013737  002174  002426  MOV FFREE,DLLSIZ        ;  AND RESTORE ORIGINAL FFREE
014256  163737  002422  002426  SUB DLLADR,DLLSIZ       ;  AND FSIZE VALUES
014264  013737  002422  002174  MOV DLLADR,FFREE
```

```
     58 014272  005726                        TST (SP)+               ;POP RETURN ADDRESS OFF STACK
     59 014274  012701  000001                MOV #1,R1               ;RUN THE DM PROGRAM AGAIN
     60 014300  004737  013134                CALL RUNDM
     61 014304  001402                        BEQ 2$
     62 014306  000137  013224                JMP RESPDM
     63 014312  000207                2$:     RETURN
```

```
                                ;T2CMD - DM REQUEST 2
                                ;
                                ;GET MANUAL INTERVENTION COMMAND
                                ;
                                ;INPUTS:
                                ;       R5 - CONTROLLER TABLE ADDRESS
                                ;       R4 - MESSAGE DATA ADDRESS
                                ;          (R4) DRIVE NUMBER
                                ;           2.(R4) OPERATION CODE
                                ;                  0 ON FIRST REQUEST FOR DRIVE. ECHO OF PREVIOUS RESPONSE ALL OTHER TIMES.
                                ;             IF OPERATION CODE = 2
                                ;              4.(R4) DATA BYTE READ (TO BE PRINTED)
                                ;       R3 - COMMAND DATA ADDRESS
                                ;OUTPUTS:
                                ;       COMMAND DATA FILLED WITH THE FOLLOWING:
                                ;          (R3) OPERATION CODE
                                ;                  0 - EXIT
                                ;                  1 - WRITE
                                ;                  2 - READ
                                ;                  3 - DIAGNOSE
                                ;             IF OPERATION CODE = 1, 2 OR 3
                                ;              2.(R3) REGION NUMBER
                                ;              4.(R3) OFFSET INTO REGION
                                ;                IF OPERATION CODE = 1
                                ;              6.(R3) DATA BYTE
                                ;       Z SET IF DATA RETURNED
                                ;       Z CLEAR IF DRIVE NUMBER NOT ON THIS CONTROLLER

014314  032737  000200  002160  T2CMD:  BIT #SM.MAN,SFPTBL+SO.BIT       ;LOOK AT MANUAL INTERVENTION MODE
014322  001002                         BNE T2CMDM                       ;EXIT IF NOT WANTED
014324  000137  015122                 JMP T2CMDX
014330                          T2CMDM: MANUAL                           ;MANUAL INTERVENTION ALLOWED?
014330  104450                                                                   TRAP    C$MANI
014332                                  BCOMPLETE T2CMD0                 ;PRINT WARNING IF NOT
014332  103412                                                                   BCS     T2CMD0
014334                          T2CMD9: PRINTF #T2WARN
014334  012746  010312                                                           MOV     #T2WARN,-(SP)
014340  012746  000001                                                           MOV     #1,-(SP)
014344  010600                                                                   MOV     SP,R0
014346  104417                                                                   TRAP    C$PNTF
014350  062706  000004                                                           ADD     #4,SP
014354  000137  015122                 JMP T2CMDX
014360  012401                  T2CMD0: MOV (R4)+,R1                     ;GET DRIVE NUMBER
014362  012402                         MOV (R4)+,R2                     ;GET OPERATION CODE
014364  001056                         BNE T2CMD2                       ;BRANCH IF NOT ZERO
014366  004737  017414                 CALL GTDRVT                      ;GET DRIVE TABLE ADDRESS
014372  001401                         BEQ 1$                           ;CHECK IF DRIVE FOUND
014374  000207                         RETURN                           ;RETURN WITH Z CLEAR IF NOT
014376                          1$:     PRINTF #T2CMS1,D.UNIT(R4),(R5),(R4)      ;PRINT DESCRIPTION
014376  011446                                                                   MOV     (R4),-(SP)
014400  011546                                                                   MOV     (R5),-(SP)
014402  016446  000002                                                           MOV     D.UNIT(R4),-(SP)
014406  012746  010414                                                           MOV     #T2CMS1,-(SP)
014412  012746  000004                                                           MOV     #4,-(SP)
014416  010600                                                                   MOV     SP,R0
014420  104417                                                                   TRAP    C$PNTF
014422  062706  000012                                                           ADD     #12,SP
```

```
    43 014426                                   PRINTF #T2CMS2
       014426  012746  010523                                        MOV     #T2CMS2,-(SP)
       014432  012746  000001                                        MOV     #1,-(SP)
       014436  010600                                                MOV     SP,R0
       014440  104417                                                TRAP    C$PNTF
       014442  062706  000004                                        ADD     #4,SP
    44 014446                                   PRINTF #T2CMS3
       014446  012746  010640                                        MOV     #T2CMS3,-(SP)
       014452  012746  000001                                        MOV     #1,-(SP)
       014456  010600                                                MOV     SP,R0
       014460  104417                                                TRAP    C$PNTF
       014462  062706  000004                                        ADD     #4,SP
    45 014466                                   PRINTF #T2CMS4
       014466  012746  010707                                        MOV     #T2CMS4,-(SP)
       014472  012746  000001                                        MOV     #1,-(SP)
       014476  010600                                                MOV     SP,R0
       014500  104417                                                TRAP    C$PNTF
       014502  062706  000004                                        ADD     #4,SP
    46 014506  005037  002376                   CLR T2WRR           ;CLEAR ALL STORAGE WORDS
    47 014512  005037  002400                   CLR T2WR0
    48 014516  005037  002402                   CLR T2DR
    49
    50 014522  022702  000002          T2CMD2:  CMP #2,R2           ;SEE IF LAST OPERATION WAS READ
    51 014526  001055                           BNE T2CMDQ          ;BRANCH IF NOT TO QUESTION
    52 014530  112737  000040  002172           MOVB #' ,ERRCHR     ;PRINT ONE SPACE
    53 014536                                   PRINTF #ERRONE,#ERRCHR
       014536  012746  002172                                        MOV     #ERRCHR,-(SP)
       014542  012746  005316                                        MOV     #ERRONE,-(SP)
       014546  012746  000002                                        MOV     #2,-(SP)
       014552  010600                                                MOV     SP,R0
       014554  104417                                                TRAP    C$PNTF
       014556  062706  000006                                        ADD     #6,SP
    54 014562  013701  002376                   MOV T2WRR,R1        ;PRINT REGION
    55 014566  004737  020420                   CALL T2PNTW
    56 014572  013701  002400                   MOV T2WR0,R1        ;PRINT OFFSET
    57 014576  004737  020420                   CALL T2PNTW
    58 014602  112737  000057  002172           MOVB #'/,ERRCHR     ;PRINT A SLASH
    59 014610                                   PRINTF #ERRONE,#ERRCHR
       014610  012746  002172                                        MOV     #ERRCHR,-(SP)
       014614  012746  005316                                        MOV     #ERRONE,-(SP)
       014620  012746  000002                                        MOV     #2,-(SP)
       014624  010600                                                MOV     SP,R0
       014626  104417                                                TRAP    C$PNTF
       014630  062706  000006                                        ADD     #6,SP
    60 014634  012401                           MOV (R4)+,R1        ;PRINT THE DATA
    61 014636  004737  020472                   CALL T2PNTB
    62 014642                                   PRINTF #ERRNL       ;END THE LINE
       014642  012746  005321                                        MOV     #ERRNL,-(SP)
       014646  012746  000001                                        MOV     #1,-(SP)
       014652  010600                                                MOV     SP,R0
       014654  104417                                                TRAP    C$PNTF
       014656  062706  000004                                        ADD     #4,SP
    63
    64                                     ;NOW ASK FOR COMMAND INPUT
    65
    66 014662                              T2CMDQ: GMANID T4OPT7,TEMP,A,-1,1,20.,NO        ;ASK FOR COMMAND
       014662  104443                                                TRAP    C$GMAN
```

```
        014664  000406                                                          BR      10000$
        014666  002260                                                          .WORD   TEMP
        014670  000142                                                          .WORD   T$CODE
        014672  004176                                                          .WORD   T4OP17
        014674  177777                                                          .WORD   -1
        014676  000001                                                          .WORD   T$LOLIM
        014700  000024                                                          .WORD   T$HILIM
        014702                                                          10000$:
    67  014702  012701  002260            MOV #TEMP,R1                ;GET POINTER TO STRING
    68  014706  112100                    MOVB (R1)+,R0               ;GET COMMAND CHARACTER
    69  014710  022700  000105            CMP #'E,R0
    70  014714  001415                    BEQ T2CMDV
    71  014716  022700  000104            CMP #'D,R0
    72  014722  001016                    BNE T2CMD3
    73  014724  012713  000003            MOV #3,(R3)                 ;STORE DIAGNOSE OPERATION CODE
    74  014730  004737  020620            CALL T2GNUM                 ;GET REGION FROM COMMAND
    75  014734  001402                    BEQ 1$
    76  014736  010437  002402            MOV R4,T2DR
    77  014742  013763  002402  000002 1$: MOV T2DR,2(R3)
    78  014750  004737  020620 T2CMDV:    CALL T2GNUM                 ;MAKE SURE AT END OF LINE
    79  014754  001064                    BNE T2CMDE
    80  014756  000461                    BR T2CMDX
    81
    82                           ;COMMAND MUST BE EITHER READ OR WRITE
    83
    84  014760  012713  000002 T2CMD3:    MOV #2,(R3)                 ;CHECK IF READ
    85  014764  022700  000122            CMP #'R,R0
    86  014770  001415                    BEQ T2CMDR
    87  014772  022700  000127            CMP #'W,R0                  ;CHECK IF WRITE
    88  014776  001053                    BNE T2CMDE                  ;  IF NOT - ERROR
    89  015000  012713  000001            MOV #1,(R3)
    90  015004  004737  020620            CALL T2GNUM                 ;GET DATA BYTE
    91  015010  001446                    BEQ T2CMDE                  ;ERROR IF NO DATA
    92  015012  162700  000002            SUB #2,R0
    93  015016  003043                    BGT T2CMDE                  ;OR GREATER THAN TWO DIGITS
    94  015020  010463  000006            MOV R4,6(R3)                ;STORE DATA BYTES IN BUFFER
    95  015024  013763  002376  000002 T2CMDR: MOV T2WRR,2(R3)        ;PUT REGION AND OFFSET
    96  015032  013763  002400  000004    MOV T2WRO,4(R3)             ;  INTO BUFFER
    97  015040  021302                    CMP (R3),R2                 ;IF SO,
    98  015042  001002                    BNE T2CMDN
    99  015044  005263  000004            INC 4(R3)                   ;  INCREMENT OFFSET
   100  015050  004737  020620 T2CMDN:    CALL T2GNUM
   101  015054  001411                    BEQ T2CMDW
   102  015056  010463  000002            MOV R4,2(R3)
   103  015062  005063  000004            CLR 4(R3)
   104  015066  004737  020620            CALL T2GNUM
   105  015072  001402                    BEQ T2CMDW
   106  015074  010463  000004            MOV R4,4(R3)
   107  015100  004737  020620 T2CMDW:    CALL T2GNUM
   108  015104  001010                    BNE T2CMDE
   109  015106  016337  000002  002376    MOV 2(R3),T2WRR             ;SAVE REGION
   110  015114  016337  000004  002400    MOV 4(R3),T2WRO             ;SAVE OFFSET
   111  015122  000264 T2CMDX:    SEZ
   112  015124  000207                    RETURN
   113  015126 T2CMDE:    PRINTF #T2CMS5                              ;REPORT ERROR MESSAGE
        015126  012746  011022                                                  MOV     #T2CMS5,-(SP)
        015132  012746  000001                                                  MOV     #1,-(SP)
```

GLOBAL SUBROUTINES SECTION

```
        015136  010600                                                  MOV     SP,R0
        015140  104417                                                  TRAP    C$PNTF
        015142  062706  000004                                          ADD     #4,SP
    114 015146  000645          BR T2CMDQ                   ;GO ASK AGAIN
```

```
                                ;T4MPRM - DM REQUEST 3
                                ;
                                ;REQUEST FOR TEST 4 CONTENTS OF DATA PATTERN 16.
                                ;
                                ;INPUTS:
                                ;       R5 - CONTROLLER TABLE ADDRESS
                                ;       R4 - MESSAGE DATA ADDRESS
                                ;            (NO DATA)
                                ;       R3 - COMMAND DATA ADDRESS
                                ;OUTPUTS:
                                ;       COMMAND DATA FILLED WITH THE FOLLOWING:
                                ;          (R3) NUMBER OF WORDS IN DATA PATTERN 16
                                ;         2.(R3) DATA IN PATTERN 16
                                ;                ..
                                ;                ..
                                ;       32.(R3)
                                ;       Z SET

015150  012701  000021          T4MPRM: MOV #17.,R1                     ;GET COUNT
015154  012702  002310                  MOV #PAT16C,R2                  ; AND ADDRESS OF PATTERN 16 PARAMETERS
015160  012223                  1$:     MOV (R2)+,(R3)+                 ;COPY THE DATA TO BUFFER
015162  005301                          DEC R1
015164  001375                          BNE 1$
015166  000264                          SEZ                             ;RETURN WITH Z SET
015170  000207                          RETURN
```

```
                        ;T4UPRM - DM REQUEST 4
                        ;
                        ;REQUEST FOR TEST 4 UNIT PARAMETERS AND CLEAR STATISTICS FOR DRIVE
                        ;
                        ;INPUTS:
                        ;       R5 - CONTROLLER TABLE ADDRESS
                        ;       R4 - MESSAGE DATA ADDRESS
                        ;          (R4) DRIVE NUMBER
                        ;        2.(R4) DRIVE SERIAL NUMBER
                        ;           .
                        ;        6.(R4)
                        ;        8.(R4) HDA SERIAL NUMBER
                        ;
                        ;       14.(R4)
                        ;       R3 - COMMAND DATA ADDRESS
                        ;OUTPUTS:
                        ;       COMMAND DATA FILLED WITH THE FOLLOWING:
                        ;          (R3) PARAMETER BITS (1 F R TRUE)
                        ;               BIT 14 - INITIAL WRI
                        ;               BIT 13 - DIAGNOSTIC C  INDERS
                        ;               BIT 12 - ECC CORRECTION
                        ;               BIT 11 - READ ONLY
                        ;               BIT 10 - WRITE ONLY
                        ;               BIT  9 - RETRIES
                        ;               BIT  8 - TRACK/GROUP AND CYLINDERS SPECIFIED
                        ;               BIT  7 - (NOT USED)
                        ;               BIT  6 - SEQUENTIAL SEEKS
                        ;               BIT  5 - BEGIN/END SETS SPECIFIED
                        ;               BIT  4 - TRACK SPECIFIED (0 - GROUPS SPECIFIED)
                        ;                        HAS MEANING ONLY WHEN BIT 5 IS ZERO
                        ;               BIT  3 - WRITE CHECKS ENABLED
                        ;               BIT  2 - WRITE CHECKS ALWAYS
                        ;               BIT  1 - DATA COMPARES ENABLED
                        ;               BIT  0 - DATA COMPARE ALWAYS
                        ;        2.(R3) DATA PATTERN NUMBER
                        ;       IF PARAMETER BIT 5 SET
                        ;        4.(R3) COUNT OF BEGIN/END SETS
                        ;        6.(R3) END BLOCK (2 WORDS) THEN BEGIN BLOCK (2 WORDS)
                        ;           :       1 TO 4 SETS
                        ;           :          OR
                        ;           :       IF COUNT OF BEGIN/END BLOCKS = 0
                        ;       36.(R3)  START CYLINDER (2 WORDS) THEN END CYLINDER (2 WORDS)
                        ;                END CYLINDER A NEGATIVE VALUE IF TO TEST ENTIRE AREA
                        ;       IF PARAMETER BIT 5 CLEAR
                        ;        4.(R3) STARTING CYLINDER
                        ;        6.(R3)   (2 WORDS)
                        ;        8.(R3) ENDING CYLINDER (2 WORDS)
                        ;       10.(R3)   NEGATIVE FOR ALL CYLINDERS
                        ;       12.(R3) NUMBER OF TRACKS OR GROUPS SPECIFIED
                        ;       14.(R3) 1 TO 7 TRACK OR GROUP NUMBERS
                        ;           :       DETERMINED BY PARAMETER BIT 4
                        ;       26.(R3)
                        ;       Z SET IF DATA RETURNED
                        ;       Z CLEAR IF UNIT NUMBER NOT ON THIS CONTROLLER
                        ;       D.XFRW, D.XFRR, D.HERR, D.SERR, D.SEEK AND D.ECC CLEARED
                        ;        IN DRIVE TABLE
```

GLOBAL SUBROUTINES SECTION

```
 1 015172  012401                  T4UPRM: MOV (R4)+,R1                    ;GET DRIVE NUMBER
 2 015174  010402                          MOV R4,R2                       ;SAVE DATA ADDRESS
 3 015176  004737  017414                  CALL GTDRVT                     ;GET DRIVE TABLE ADDRESS
 4 015202  001145                          BNE T4UPRE                      ;CHECK IF DRIVE FOUND
 5 015204  012264  000200                  MOV (R2)+,D.SERN(R4)            ;COPY DRIVE SERIAL NUMBER TO DRIVE TABLE
 6 015210  012264  000202                  MOV (R2)+,D.SERN+2(R4)
 7 015214  012264  000204                  MOV (R2)+,D.SERN+4(R4)
 8 015220  016401  000004                  MOV D.PRM(R4),R1                ;GET PARAMETER BITS
 9 015224  042701  140200                  BIC #D.ZERO,R1                  ;CLEAR SOME BITS
10 015230  032737  000020  002224          BIT #ISTRTH,IFLAGS              ;IF FIRST TIME TEST 4 BEING RUN
11 015236  001406                          BEQ 2$                          ;  AFTER A START COMMAND
12 015240  032737  040000  002160          BIT #SM.IW,SFPTBL+SO.BIT        ;GET INITIAL WRITE BIT
13 015246  001402                          BEQ 2$
14 015250  052701  040000                  BIS #D.IW,R1                    ;MOVE INTO PARAMETER BITS
15 015254  010123                  2$:     MOV R1,(R3)+                    ;PUT INTO BUFFER
16 015256  016423  000006                  MOV D.PAT(R4),(R3)+             ;PUT PATTERN NUMBER IN BUFFER
17 015262  032701  000040                  BIT #D.BE,R1                    ;CHECK BEGIN/END PARAMETER BIT
18 015266  001421                          BEQ 10$                         ;BRANCH IF NOT SET
19
20                                    ;RETURN BEGIN/END SETS
21
22 015270  012701  000004                  MOV #4,R1                       ;GET COUNT OF SETS
23 015274  010402                          MOV R4,R2                       ;GET INDEX INTO DRIVE TABLE
24 015276  062702  000112                  ADD #D.BEC,R2
25 015302  012223                          MOV (R2)+,(R3)+                 ;COPY COUNT
26 015304  016223  000004          1$:     MOV 4(R2),(R3)+
27 015310  016223  000006                  MOV 6(R2),(R3)+
28 015314  012223                          MOV (R2)+,(R3)+
29 015316  012223                          MOV (R2)+,(R3)+
30 015320  062702  000004                  ADD #4,R2
31 015324  005301                          DEC R1
32 015326  001366                          BNE 1$
33 015330  000456                          BR T4UPRX
34
35 015332  032764  000400  000004  10$:     BIT #D.CYL,D.PRM(R4)            ;LOOK AT D.CYL BIT
36 015336  001441                          BEQ 20$                         ;BRANCH IF NOT SET
37
38                                    ;RETURN TRACKS/GROUPS AND CYLINDERS
39
40 015342  005764  000112                  TST D.BEC(R4)                   ;CHECK IF ANY TRACKS/GROUPS
41 015346  001421                          BEQ 25$                         ;BRANCH IF NONE
42 015350  012701  000004                  MOV #4,R1                       ;COUNT OF CYLINDER WORDS
43 015354  010402                          MOV R4,R2
44 015356  062702  000154                  ADD #D.BCYL,R2
45 015362  012223                  11$:    MOV (R2)+,(R3)+                 ;CYLINDERS
46 015364  005301                          DEC R1
47 015366  001375                          BNE 11$
48 015370  012701  000010                  MOV #8.,R1
49 015374  010402                          MOV R4,R2
50 015376  062702  000112                  ADD #D.BEC,R2
51 015402  012223                  12$:    MOV (R2)+,(R3)+                 ;TRACKS/GROUPS
52 015404  005301                          DEC R1
53 015406  001375                          BNE 12$
54 015410  000426                          BR T4UPRX
55
56                                    ;RETURN CYLINDERS ONLY
57
```

```
     58 015412  052763  000040  177774  25$:    BIS #D.BE,-4(R3)          ;SET D.BE FOR DM PROGRAM
     59 015420  005023                          CLR (R3)+                 ;SEND ZERO BEGIN/END COUNT
     60 015422  012701  000004                  MOV #4,R1
     61 015426  010402                          MOV R4,R2
     62 015430  062702  000154                  ADD #D.BCYL,R2
     63 015434  012223                  26$:    MOV (R2)+,(R3)+           ;CYLINDERS
     64 015436  005301                          DEC R1
     65 015440  001375                          BNE 26$
     66 015442  000411                          BR T4UPRX
     67
     68                                 ;RETURN ENTIRE AREA
     69
     70 015444  052763  000040  177774  20$:    BIS #D.BE,-4(R3)          ;SET D.BE FOR DM PROGRAM
     71 015452  005023                          CLR (R3)+                 ;BEGIN/END COUNT OF ZERO
     72 015454  005023                          CLR (R3)+                 ;START CYLINDER OF ZERO
     73 015456  005023                          CLR (R3)+
     74 015460  005023                          CLR (R3)+                 ;END CYLINDER NEGATIVE
     75 015462  012723  177777                  MOV #-1,(R3)+
     76 015466  005064  000164          T4UPRX: CLR D.XFRW(R4)            ;CLEAR STATISTICS
     77 015472  005064  000166                  CLR D.XFRR(R4)
     78 015476  005064  000170                  CLR D.HERR(R4)
     79 015502  005064  000172                  CLR D.SERR(R4)
     80 015506  005064  000174                  CLR D.SEEK(R4)
     81 015512  005064  000176                  CLR D.ECCC(R4)
     82 015516  000207                  T4UPRE: RETURN
```

```
                                        ;T4BB1 - DM REQUEST 5
                                        ;
                                        ;REQUEST FOR FIRST 14 BAD BLOCKS
                                        ;
                                        ;INPUTS:
                                        ;       R5 - CONTROLLER TABLE ADDRESS
                                        ;       R4 - MESSAGE DATA ADDRESS
                                        ;          (R4) DRIVE NUMBER
                                        ;       R3 - COMMAND DATA ADDRESS
                                        ;OUTPUTS:
                                        ;       COMMAND DATA FILLED WITH BAD BLOCKS
                                        ;          (R3) COUNT OF BAD BLOCKS
                                        ;           2.(R3) BAD BLOCK 1 (LOW)
                                        ;           4.(R3)             (HIGH)
                                        ;              :
                                        ;              :
                                        ;           56.(R3) BAD BLOCK 14 (LOW)
                                        ;           58.(R3)              (HIGH)
                                        ;       Z SET IF DATA RETURNED
                                        ;       Z CLEAR IF DRIVE NUMBER NOT ON THIS CONTROLLER

015520  011401              T4BB1:  MOV (R4),R1             ;GET DRIVE NUMBER
015522  004737  017414              CALL GTDRVT             ;GET DRIVE TABLE ADDRESS
015526  001007                      BNE T4BB1E              ;CHECK IF DRIVE FOUND
015530  062704  000010              ADD #D.BB,R4            ;INCREASE ADDRESS TO DATA TO COPY
015534  012701  000035              MOV #<1+<14.*2>>,R1     ;GET COUNT OF WORDS
015540  012423              1$:     MOV (R4)+,(R3)+         ;COPY THE WORDS
015542  005301                      DEC R1
015544  001375                      BNE 1$
015546  000207              T4BB1E: RETURN
```

```
                                        ;T4BB2 - DM REQUEST 6
                                        ;
                                        ;REQUEST LAST TWO BAD BLOCKS
                                        ;
                                        ;INPUTS:
                                        ;       R5 - CONTROLLER TABLE ADDRESS
                                        ;       R4 - MESSAGE DATA ADDRESS
                                        ;          (R4) DRIVE NUMBER
                                        ;       R3 - COMMAND DATA ADDRESS
                                        ;OUTPUTS:
                                        ;       COMMAND DATA FILLED WITH BAD BLOCKS 15 AND 16
                                        ;       Z SET IF DATA RETURNED
                                        ;       Z CLEAR IF UNIT NUMBER NOT ON THIS CONTROLLER
                                        ;
015550  011401                  T4BB2:  MOV (R4),R1                     ;GET DRIVE NUMBER
015552  004737  017414                  CALL GTDRVT                     ;GET DRIVE TABLE ADDRESS
015556  001007                          BNE T4BB2E                      ;CHECK IF DRIVE FOUND
015560  062704  000102                  ADD #D.BB15,R4                  ;INCREASE ADDRESS TO DATA TO COPY
015564  012701  000004                  MOV #4,R1                       ;GET COUNT OF WORDS
015570  012423                  1$:     MOV (R4)+,(R3)+                 ;COPY THE WORDS
015572  005301                          DEC R1
015574  001375                          BNE 1$
015576  000207                  T4BB2E: RETURN
```

```
                                        ;T4SOFT - DM REQUEST 7
                                        ;
                                        ;ADD TO SOFT ERROR AND ECC COUNTS
                                        ;
                                        ;INPUTS:
                                        ;       R5 - CONTROLLER TABLE ADDRESS
                                        ;       R4 - MESSAGE DATA ADDRESS
                                        ;          (R4) DRIVE NUMBER
                                        ;           2.(R4) VALUE TO ADD TO SOFT ERROR COUNT
                                        ;           4.(R4) VALUE TO ADD TO ECC COUNT
                                        ;       R3 - COMMAND DATA ADDRESS
                                        ;
015600  012401                  T4SOFT: MOV (R4)+,R1                    ;GET DRIVE NUMBER
015602  010402                          MOV R4,R2                       ;SAVE DATA ADDRESS
015604  004737  017414                  CALL GTDRVT                     ;GET DRIVE TABLE ADDRESS
015610  001005                          BNE 1$                          ;CHECK IF DRIVE FOUND
015612  062264  000172                  ADD (R2)+,D.SERR(R4)            ;ADD TO SOFT ERROR COUNT
015616  062264  000176                  ADD (R2)+,D.ECCC(R4)            ;ADD TO ECC COUNT
015622  000264                          SEZ
015624  000207                  1$:     RETURN
```

GLOBAL SUBROUTINES SECTION

```
 1 015626                           T4SEEK:
 2                                  ;       DM REQUEST 8.
 3                                  ;
 4                                  ;RECORD 1000 SEEKS COMPLETED ON DRIVE
 5                                  ;
 6                                  ;INPUTS:
 7                                  ;       R5 - CONTROLLER TABLE ADDRESS
 8                                  ;       R4 - MESSAGE DATA ADDRESS
 9                                  ;          (R4) DRIVE NUMBER
10                                  ;       R3 - COMMAND DATA ADDRESS
11                                  ;
12 015626  011401                           MOV     (R4),R1             ; GET DRIVE NUMBER
13 015630  004737  017414                   CALL    GTDRVT              ; GET DRIVE TABLE ADDRESS
14 015634  001003                           BNE     SEKERE              ; CHECK IF DRIVE FOUND
15 015636  005264  000174                   INC     D.SEEK(R4)          ; COUNT THE BITS TRANSFERRED
16 015642  000264                          SEZ                         ; NORMAL RETURN
17 015644  000207                  SEKERE: RETURN
```

```
                                        ;T4MXFR - DM REQUEST 9.
                                        ;
                                        ;RECORD 1M BITS TRANSFERRED ON UNIT. COMPARE TO TRANSFER LIMIT AND
                                        ;REPORT LIMIT REACHED.
                                        ;
                                        ;INPUTS:
                                        ;       R5 - CONTROLLER TABLE ADDRESS
                                        ;       R4 - MESSAGE DATA ADDRESS
                                        ;          (R4) DRIVE NUMBER
                                        ;           2.(R4) VALUE TO ADD TO READ COUNT
                                        ;           4.(R4) VALUE TO ADD TO WRITE COUNT
                                        ;       R3 - COMMAND DATA ADDRESS
                                        ;OUTPUTS:
                                        ;          (R3) BIT 15 SET IF TRANSFER LIMIT REACHED
                                        ;       MESSAGE PRINTED IF TRANSFER LIMIT REACHED
                                        ;       Z CLEAR IF DRIVE NUMBER NOT ON THIS CONTROLLER
                                        ;
015646  010402                  T4MXFR: MOV R4,R2                       ;GET MESSAGE DATA ADDRESS
015650  011401                          MOV (R4),R1                     ;GET DRIVE NUMBER
015652  004737  017414                  CALL GTDRVT                     ;GET DRIVE TABLE ADDRESS
015656  001053                          BNE MXFERE                      ;CHECK IF DRIVE FOUND
015660  066264  000002  000166          ADD 2(R2),D.XFRR(R4)            ;ADD MEGABITS READ
015666  066264  000004  000164          ADD 4(R2),D.XFRW(R4)            ;ADD MEGABITS WRITTEN
015674  005737  002156                  TST SFPTBL+SO.XL                ;SEE IF LIMIT SPECIFIED
015700  001441                          BEQ MXFERX                      ;BRANCH IF NOT
015702  026437  000166  002156          CMP D.XFRR(R4),SFPTBL+SO.XL     ;CHECK IF LIMIT REACHED
015710  103435                          BLO MXFERX                      ;BRANCH IF LIMIT NOT REACHED
015712                                  RFLAGS R0                       ;CHECK FLAGS
015712  104421                                                                  TRAP    C$RFLA
015714  032700  000040                  BIT #IDU,R0                     ;SEE IF DROPPING UNITS IS INHIBITED
015720  001031                          BNE MXFERX
015722  052713  100000                  BIS #BIT15,(R3)                 ;SET DROP UNIT BIT
015726  042765  000010  000014          BIC #CT.MSG,C.FLG(R5)           ;CLEAR MESSAGE RECEIVED FLAG
015734                                  PRINTF #MESSG,D.UNIT(R4),(R5),(R4)      ;PRINT TESTING DONE
015734  011446                                                                  MOV     (R4),-(SP)
015736  011546                                                                  MOV     (R5),-(SP)
015740  016446  000002                                                          MOV     D.UNIT(R4),-(SP)
015744  012746  010131                                                          MOV     #MESSG,-(SP)
015750  012746  000004                                                          MOV     #4,-(SP)
015754  010600                                                                  MOV     SP,R0
015756  104417                                                                  TRAP    C$PNTF
015760  062706  000012                                                          ADD     #12,SP
015764                                  PRINTF #MXFERP
015764  012746  006425                                                          MOV     #MXFERP,-(SP)
015770  012746  000001                                                          MOV     #1,-(SP)
015774  010600                                                                  MOV     SP,R0
015776  104417                                                                  TRAP    C$PNTF
016000  062706  000004                                                          ADD     #4,SP
016004  000264                  MXFERX: SEZ                             ;NORMAL RETURN
016006  000207                  MXFERE: RETURN
```

```
                                ;UTOTST - DM REQUEST 10
                                ;
                                ;TELL DM PROGRAM WHICH DRIVES ARE SELECTED FOR TESTING
                                ;
                                ;INPUTS:
                                ;       R5 - CONTROLLER TABLE ADDRESS
                                ;       R4 - MESSAGE DATA ADDRESS
                                ;            (NO DATA)
                                ;       R3 - COMMAND DATA ADDRESS
                                ;OUTPUTS:
                                ;       COMMAND PACKET CONTAINING UP TO 8 DRIVE NUMBERS.
                                ;         LIST IS ENDED BY A WORD WITH BIT 15 SET.
                                ;         FOLLOWING LIST IS THE INFORMATION FROM T2DLL REQUEST IF APPLICABLE.
                                ;       Z SET
                                
016010  010504          UTOTST: MOV R5,R4                       ;GET ADDRESS OF CONTROLLER TABLE
016012  062704  000020          ADD #C.DRO,R4                   ;BUMP TO DRIVE TABLE POINTERS
016016  012702  000010          MOV #8.,R2                      ;GET COUNT OF PORTS
016022  012400          UTOT1:  MOV (R4)+,R0                    ;SEE IF DRIVE TABLE POINTER EXISTS
016024  001406                  BEQ UTOT2                       ;BRANCH IF NOT
016026  005760  000002          TST D.UNIT(R0)                  ;LOOK IF UNIT AVAILABLE FOR TESTING
016032  100773                  BMI UTOT1
016034                              ASSUME DT.AVL EQ BIT15
016034  011023                  MOV (R0),(R3)+                  ;LOAD DRIVE NUMBER FROM TABLE
016036  005302                  DEC R2                          ;COUNT THE PORTS
016040  001370                  BNE UTOT1                       ;REPEAT FOR EACH PORT
016042  012723  100000  UTOT2:  MOV #BIT15,(R3)+                ;TERMINATE LIST
016046  013723  002412          MOV DLL,(R3)+                   ;GET DLL CONTROL WORD
016052  001407                  BEQ UTOT4                       ; IF NON-ZERO
016054  012701  002414          MOV #DLLDR,R1                   ;  TRANSFER ALL DLL WORDS INTO BUFFER
016060  012702  000020          MOV #<DLLNAM+4-DLLDR>,R2
016064  012123          UTOT3:  MOV (R1)+,(R3)+
016066  005302                  DEC R2
016070  001375                  BNE UTOT3
016072  000264          UTOT4:  SEZ
016074  000207                  RETURN                          ;RETURN WITH Z SET
```

```
                                ;ERRMES  - DM REQUEST 11
                                ;
                                ;PRINT AN ERROR MESSAGE
                                ;
                                ;INPUTS:
                                ;       R5 - CONTROLLER TABLE ADDRESS
                                ;       R4 - MESSAGE DATA ADDRESS
                                ;            (R4)  ERROR PC IN DM PROGRAM
                                ;          2.(R4)  < 9:8 > ERROR TYPE
                                ;                  < 7:0 > ERROR NUMBER
                                ;          4.(R4)  DRIVE NUMBER (-1 IF NOT GIVEN)
                                ;          6.(R4)  MESSAGE POINTER
                                ;          8.(R4)  OPTIONAL PARAMETERS FOR ERROR PRINT ROUTINE
                                ;         10.(R4)                  ''
                                ;                                  ''
                                ;                                  ''
                                ;         58.(R4)                  ''
                                ;       R3 - COMMAND DATA ADDRESS
                                ;OUTPUTS:
                                ;       COMMAND PACKET CONTAINING THE FOLLOWING:
                                ;            (R3) - BIT 15 SET IF FATAL ERROR TO INDICATE DRIVE SHOULD NO LONGER BE TESTED
                                ;       Z SET TO INDICATE DATA RETURNED
                                ;       Z CLEAR IF DRIVE NUMBER NOT ON THIS CONTROLLER
                                
016076  122764  000001  000003  ERRMES: CMPB #1,3(R4)                  ;CHECK IF FATAL ERROR
016104  100406                         BMI 4$                         ;BRANCH IF NOT
016106                                 RFLAGS R0                      ;LOOK AT FLAGS
016106  104421                                                                TRAP    C$RFLA
016110  032700  000040                 BIT #IDU,R0                    ;SEE IF ALLOWED TO DROP UNITS
016114  001012                         BNE 3$                         ;BRANCH IF NOT
016116  052713  100000                 BIS #BIT15,(R3)                ;SET DROP DRIVE BIT
016122  122764  000003  000003  4$:    CMPB #3,3(R4)                  ;SEE IF SOFT ERROR
016130  001004                         BNE 3$                         ;BRANCH IF NOT
016132  032737  000400  002160         BIT #SM.SSF,SO.BIT+SFPTBL      ;SEE IF SOFT ERRORS SUPPRESSED
016140  001077                         BNE ERRMSX                     ;DON'T PRINT IF SO
016142  042765  000010  000014  3$:    BIC #CT.MSG,C.FLG(R5)          ;CLEAR MESSAGE RECEIVED FLAG
016150  022737  000004  002226         CMP #4,TNUM                    ;IF TEST # 4,
016156  001004                         BNE 1$
016160  032737  001000  002160         BIT #SM.LOG,SFPTBL+SO.BIT      ;  SEE IF LOG BEING USED
016166  001003                         BNE ERRMSL
016170  004737  020746         1$:     CALL PNTERR                    ;IF NOT, PRINT THE ERROR MESSAGE
016174  000461                         BR ERRMSX
                                
016176  005737  002404         ERRMSL: TST LBUFS                      ;SEE IF LOG BUFFER ESTABLISHED
016202  001014                         BNE 1$                         ;  LBUFS CONTAINS ADDRESS IF ESTABLISHED
016204  013701  002174                 MOV FFREE,R1                   ;GET START ADDRESS OF BUFFER
016210  010137  002404                 MOV R1,LBUFS                   ;INITIALIZE BUFFER STORAGE
016214  010137  002406                 MOV R1,LBUFN                   ;SAVE ADDRESS WHERE TO ADD
016220  063701  002176                 ADD FSIZE,R1                   ;COMPUTE END OF STORAGE AREA
016224  063701  002176                 ADD FSIZE,R1
016230  010137  002410                 MOV R1,LBUFE
016234  013701  002406         1$:     MOV LBUFN,R1                   ;GET ADDRESS OF DATA STORAGE AREA
016240  062737  000074  002406         ADD #HC.BSZ,LBUFN              ;ADD BYTES OF STORAGE NEEDED
016246  023737  002406  002410         CMP LBUFN,LBUFE                ;SEE IF ENOUGH ROOM
016254  103017                         BHIS 3$                        ;  BRANCH IF NOT
016256  010521                         MOV R5,(R1)+                   ;STORE CONTROLLER TABLE ADDRESS
016260  012700  000035                 MOV #<HC.BSZ-2>/2,R0           ;GET COUNT OF REST OF DATA IN WORDS
```

```
     57 016264  012421              2$:     MOV (R4)+,(R1)+             ;STORE DATA
     58 016266  005300                      DEC R0
     59 016270  001375                      BNE 2$
     60 016272                              PRINTF #LOGDAT              ;TELL OPERATOR DATA IN BUFFER
        016272  012746  011514                                                  MOV     #LOGDAT,-(SP)
        016276  012746  000001                                                  MOV     #1,-(SP)
        016302  010600                                                          MOV     SP,R0
        016304  104417                                                          TRAP    C$PNTF
        016306  062706  000004                                                  ADD     #4,SP
     61 016312  000412                      BR ERRMSX
     62 016314  010137  002406        3$:   MOV R1,LBUFN                ;RESTORE OLD VALUE OF LBUFN
     63 016320                              PRINTF #LOGFUL              ;TELL OPERATOR LOG IS FULL
        016320  012746  011562                                                  MOV     #LOGFUL,-(SP)
        016324  012746  000001                                                  MOV     #1,-(SP)
        016330  010600                                                          MOV     SP,R0
        016332  104417                                                          TRAP    C$PNTF
        016334  062706  000004                                                  ADD     #4,SP
     64 016340  000264              ERRMSX: SEZ
     65 016342  000207                      RETURN
```

```
                                        ;ERRMC - DM REQUEST 12.
                                        ;
                                        ;REPORT AN ERROR MESSAGE IDENTICAL TO DM REQUEST ERRMES
                                        ;THEN ADD ONE TO THE ERROR COUNT FOR THE DRIVE AND SEE IF
                                        ;ERROR LIMIT REACHED.
                                        ;
                                        ;INPUTS:
                                        ;       R5 - CONTROLLER TABLE ADDRESS
                                        ;       R4 - MESSAGE DATA ADDRESS
                                        ;            (R4)  ERROR PC IN DM PROGRAM
                                        ;          2.(R4)  < 9:8 > ERROR TYPE
                                        ;                  < 7:0 > ERROR NUMBER
                                        ;          4.(R4)  DRIVE NUMBER (-1 IF NOT GIVEN)
                                        ;          6.(R4)  MESSAGE POINTER
                                        ;          8.(R4)  OPTIONAL PARAMETERS FOR ERROR PRINT ROUTINE
                                        ;         10.(R4)                  ..
                                        ;                                  ..
                                        ;                                  ..
                                        ;         58.(R4)                  ..
                                        ;       R3 - COMMAND DATA ADDRESS
                                        ;OUTPUTS:
                                        ;       COMMAND PACKET CONTAINING THE FOLLOWING:
                                        ;         (R3) BIT 15 SET IF ERROR COUNT REACHED
                                        ;              TO INDICATE DRIVE SHOULD NO LONGER BE TESTED.
                                        ;       Z CLEAR IF DRIVE NUMBER NOT ON THIS CONTROLLER
                                        ;       Z SET TO INDICATE DATA RETURNED
                                        ;
016344                          ERRMC:  PUSH R4
016344  010446                                 MOV R4,-(SP)
016346  004737  016076                 CALL    ERRMES                  ; CALL REQUEST ERRMES
016352  001042                         BNE 4$
016354                                 POP R4
016354  012604                                 MOV (SP)+,R4
016356  016401  000004                 MOV     4(R4),R1                ; GET DRIVE NUMBER
016362  116402  000003                 MOVB 3(R4),R2                   ;GET ERROR TYPE
016366  004737  017414                 CALL    GTDRVT                  ; GET DRIVE TABLE
016372  001033                         BNE     5$                      ; EXIT IF NO TABLE FOR UNIT
016374  022702  000002                 CMP #2,R2                       ;CHECK IF HARD ERROR
016400  001025                         BNE 3$                          ;BRANCH IF NOT
016402  005264  000170                 INC     D.HERR(R4)              ; COUNT THE ERROR
016406  026437  000170  002154         CMP     D.HERR(R4),SFPTBL+SO.EL ; CHECK IF AT LIMIT
016414  103417                         BLO     3$                      ; IF LIMIT REACHED, BRANCH
016416                                 RFLAGS R0                       ;LOOK AT THE FLAGS
016416  104421                                                                 TRAP    C$RFLA
016420  032700  000040                 BIT #IDU,R0                     ;SEE IF DROPPING UNITS INHIBITED
016424  001013                         BNE 3$                          ;BRANCH IF SO
016426                                 PRINTF  #ERRLIM,R1              ; PRINT LIMIT REACHED
016426  010146                                                                 MOV     R1,-(SP)
016430  012746  006502                                                         MOV     #ERRLIM,-(SP)
016434  012746  000002                                                         MOV     #2,-(SP)
016440  010600                                                                 MOV     SP,R0
016442  104417                                                                 TRAP    C$PNTF
016444  062706  000006                                                         ADD     #6,SP
016450  052713  100000                 BIS #BIT15,(R3)                 ;SET STOP TESTING BIT
016454  000264                 3$:     SEZ                             ; SET Z FOR NORMAL RETURN
016456  000207                         RETURN                          ; RETURN TO CALLING PROGRAM
```

GLOBAL SUBROUTINES SECTION

```
     49 016460                    4$:    POP     R4                      ; ADJUST STACK
      . 016460  012604                    MOV (SP)+,R4
     50 016462  000244            5$:    CLZ                             ; FLAG AS ERROR
     51 016464  000207                   RETURN                          ; RETURN TO CALLING PROGRAM
```

```
                                ;MESSAG - DM REQUEST 13.
                                ;
                                ;PRINT A MESSAGE WITH HEADER AS FOLLOWS:
                                ;  UNIT XX UDA AT XXXXXX DRIVE XXX
                                ;
                                ;INPUTS:
                                ;       R5 - CONTROLLER TABLE ADDRESS
                                ;       R4 - MESSAGE DATA ADDRESS
                                ;          (R4) DRIVE NUMBER
                                ;        2.(R4) MESSAGE POINTER
                                ;        4.(R4) OPTIONAL MESSAGE PARAMETERS
                                ;           :
                                ;           :
                                ;       58.(R4) COMMAND DATA ADDRESS
                                ;
016466  042765  000010  000014  MESSAG: BIC #CT.MSG,C.FLG(R5)          ;CLEAR MESSAGE RECEIVED FLAG
016474  012401                          MOV (R4)+,R1                  ;GET DRIVE NUMBER
016476                                  PUSH R4                       ;SAVE DATA POINTER
016476  010446                                  MOV R4,-(SP)
016500  004737  017414                  CALL GTDRVT                   ;GET DRIVE TABLE ADDRESS
016504  001032                          BNE 1$                        ;CHECK IF DRIVE FOUND
016506                                  PRINTX #MESSG,D.UNIT(R4),(R5),(R4)      ;PRINT HEADER
016506  011446                                                          MOV     (R4),-(SP)
016510  011546                                                          MOV     (R5),-(SP)
016512  016446  000002                                                  MOV     D.UNIT(R4),-(SP)
016516  012746  010131                                                  MOV     #MESSG,-(SP)
016522  012746  000004                                                  MOV     #4,-(SP)
016526  010600                                                          MOV     SP,R0
016530  104415                                                          TRAP    C$PNTX
016532  062706  000012                                                  ADD     #12,SP
016536                                  POP R4
016536  012604                                  MOV (SP)+,R4
016540  012402                          MOV (R4)+,R2                  ;GET MESSAGE POINTER
016542  006302                          ASL R2                        ;DOUBLE TO MAKE BYTE OFFSET
016544  063702  002212                  ADD DMPROG,R2                 ;ADD TO START OF MESSAGE STRINGS
016550  067702  163436                  ADD @DMPROG,R2                ;ADD SIZE OF MAIN PROGRAM
016554  105712                          TSTB (R2)                     ;CHECK FIRST BYTE
016556  001001                          BNE 2$                        ;IF ZERO
016560  005202                          INC R2                        ;  INCREMENT TO NEXT BYTE
016562  004737  016602          2$:     CALL OSTRNG                   ;OUTPUT ACCORDING TO STRING
016566  000264                          SEZ
016570  000207                          RETURN
                                ;
016572                          1$:     POP R4
016572  012604                                  MOV (SP)+,R4
016574  000207                          RETURN
```

```
                                ;DONE - DM REQUEST 14
                                ;
                                ;MARK DM PROGRAM AS NO LONGER RUNNING
                                ;
                                ;INPUTS:
                                ;       R5 - CONTROLLER TABLE ADDRESS
                                ;       R4 - MESSAGE DATA ADDRESS
                                ;            (NO DATA)
                                ;       R3 - COMMAND DATA ADDRESS
                                ;OUTPUTS:
                                ;       Z CLEAR TO DROP UNIT FROM TESTING
                                
016576  000244                  DONE:   CLZ                             ;DROP UNIT FROM TESTING
016600  000207                          RETURN
```

```
      1                                 ;OSTRNG
      2                                 ;
      3                                 ;OUTPUT A MESSAGE ACCORDING TO A FORMAT STRING
      4                                 ;
      5                                 ;INPUTS:
      6                                 ;       R2 - ADDRESS OF START OF FORMAT STRING
      7                                 ;       R4 - ADDRESS OF PARAMETERS
      8                                 ;OUTPUTS:
      9                                 ;       R2 AND R4 UPDATED TO END OF STRING AND PARAMETERS
     10
     11 016602  112201                  OSTRNG: MOVB (R2)+,R1                   ;GET CONTROL CHARACTER
     12 016604  001425                          BEQ OSTRE                       ;EXIT IF NULL CHARACTER
     13 016606  012700  017216                  MOV #ERRC,R0                    ;GET POINTER TO CHARACTER TABLE
     14 016612  120110                  NCONS:  CMPB R1,(R0)                    ;COMPARE CHARACTER WITH TABLE ENTRY
     15 016614  001413                          BEQ NCONF                       ;BRANCH IF MATCH FOUND
     16 016616  105720                          TSTB (R0)+                      ;INCREMENT POINTER
     17 016620  001374                          BNE NCONS                       ;CONTINUE SEARCH IF NOT END OF TABLE
     18 016622                                  PRINTF #ERRME1                  ;REPORT BAD CONTROL CHARACTER
        016622  012746  006332                                                          MOV     #ERRME1,-(SP)
        016626  012746  000001                                                          MOV     #1,-(SP)
        016632  010600                                                                  MOV     SP,R0
        016634  104417                                                                  TRAP    C$PNTF
        016636  062706  000004                                                          ADD     #4,SP
     19 016642  000406                          BR OSTRE
     20 016644  162700  017216          NCONF:  SUB #ERRC,R0                    ;GET INCREMENT INTO TABLE
     21 016650  006300                          ASL R0                          ;DOUBLE TO WORD COUNT
     22 016652  004770  017230                  CALL @ERRD(R0)                  ;DISPATCH TO PRINT ROUTINE
     23 016656  000751                          BR OSTRNG                       ;GET NEXT
     24 016660  000207                  OSTRE:  RETURN
```

```
                                        ;CONTROL CHARACTER WAS A QUOTE. PRINT ALL CHARACTERS TO THE NEXT QUOTE.

016662  112237  002172                CON.QU: MOVB (R2)+,ERRCHR               ;GET CHARACTER
016666  123727  002172  000042                CMPB ERRCHR,#'''                ;CHECK IF ENDING QUOTE
016674  001413                              BEQ CON.QX                      ;IF SO, GO GET NEXT CONTROL CHARACTER
016676                                      PRINTX #ERRONE,#ERRCHR          ;PRINT THE CHARACTER
016676  012746  002172                                                                MOV     #ERRCHR,-(SP)
016702  012746  005316                                                                MOV     #ERRONE,-(SP)
016706  012746  000002                                                                MOV     #2,-(SP)
016712  010600                                                                      MOV     SP,R0
016714  104415                                                                      TRAP    C$PNTX
016716  062706  000006                                                              ADD     #6,SP
016722  000757                              BR CON.QU                       ;CONTINUE PRINTING
016724  000207                          CON.QX: RETURN

                                        ;CONTROL CHARACTER WAS AN A. PRINT ASCII CHARACTERS FROM PARAMETERS.

016726  004737  017504                CON.A:  CALL GETCNT                     ;GET COUNT OF CHARACTERS
016732  112437  002172                CON.A1: MOVB (R4)+,ERRCHR               ;GET CHARACTER
016736                                      PRINTX #ERRONE,#ERRCHR          ;PRINT THE CHARACTER
016736  012746  002172                                                                MOV     #ERRCHR,-(SP)
016742  012746  005316                                                                MOV     #ERRONE,-(SP)
016746  012746  000002                                                                MOV     #2,-(SP)
016752  010600                                                                      MOV     SP,R0
016754  104415                                                                      TRAP    C$PNTX
016756  062706  000006                                                              ADD     #6,SP
016762  005301                              DEC R1                          ;COUNT THE CHARACTERS
016764  001362                              BNE CON.A1                      ;PRINT UNTIL COUNT REACHES ZERO
016766  032704  000001                      BIT #1,R4                       ;CHECK IF R4 NOW ODD
016772  001401                              BEQ CON.A2
016774  005204                              INC R4                          ;IF SO, INCREMENT TO NEXT EVEN ADDRESS
016776  000207                          CON.A2: RETURN                      ;NOW GET NEXT CONTROL CHARACTER

                                        ;CONTROL CHARACTER WAS A D. PRINT DECIMAL NUMBER.

017000  012701  000012                CON.D:  MOV #10.,R1                     ;LOAD RADIX
017004  004737  017562                        CALL PNTNUM                     ;PRINT NUMBER
017010  000207                              RETURN                          ;NOW GET NEXT CONTROL CHARACTER

                                        ;CONTROL CHARACTER WAS AN H. PRINT HEX NUMBER.

017012  012701  000020                CON.H:  MOV #16.,R1                     ;LOAD RADIX
017016  004737  017562                        CALL PNTNUM                     ;PRINT NUMBER
017022  000207                              RETURN                          ;NOW GET NEXT CONTROL CHARACTER

                                        ;CONTROL CHARACTER WAS AN O. PRINT OCTAL NUMBEP.

017024  012701  000010                CON.O:  MOV #8.,R1                      ;LOAD RADIX
017030  004737  017562                        CALL PNTNUM                     ;PRINT NUMBER
017034  000207                              RETURN                          ;NOW GET NEXT CONTROL CHARACTER

                                        ;CONTROL CHARACTER WAS AN N. PRINT NEW LINE SEQUENCE.

017036  004737  017504                CON.N:  CALL GETCNT                     ;GET COUNT
017042                                  CON.N1: PRINTX #ERRNL                   ;PRINT NEW LINE SEQUENCE
017042  012746  005321                                                                MOV     #ERRNL,-(SP)
017046  012746  000001                                                                MOV     #1,-(SP)
```

```
017052  010600                                                          MOV     SP,R0
017054  104415                                                          TRAP    C$PNTX
017056  062706  000004                                                  ADD     #4,SP
 44 017062  005301                  DEC R1                              ;COUNT THE SEQUENCES
 45 017064  001366                  BNE CON.N1
 46 017066  000207                  RETURN                              ;NOW GET NEXT CONTROL CHARACTER
 47
 48                         ;CONTROL CHARACTER WAS AN R. CALL A PRE-PROGRAMMED ROUTINE.
 49
 50 017070  004737  017504  CON.R:  CALL GETCNT                         ;GET ROUTINE NUMBER
 51 017074  020127  000002          CMP R1,#ERRRSZ                      ;CHECK IF DEFINED ROUTINE NUMBER
 52 017100  101004                  BHI CON.R1
 53 017102  060101                  ADD R1,R1                           ;DOUBLE COUNT TO GET WORD INDEX
 54 017104  004771  017200          CALL @ERRRTB-2(R1)                  ;CALL ROUTINE
 55 017110  000207                  RETURN                              ;NOW GET NEXT CONTROL CHARACTER
 56 017112          CON.R1: PRINTF #ERRME1                              ;REPORT BAD MESSAGE STRING
017112  012746  006332                                                  MOV     #ERRME1,-(SP)
017116  012746  000001                                                  MOV     #1,-(SP)
017122  010600                                                          MOV     SP,R0
017124  104417                                                          TRAP    C$PNTF
017126  062706  000004                                                  ADD     #4,SP
 57 017132                  POP R1                                      ;FIX THE STACK
017132  012601                      MOV (SP)+,R1
 58 017134  000207                  RETURN
 59
 60                         ;CONTROL CHARACTER WAS AN S. PRINT SPACES.
 61
 62 017136  004737  017504  CON.S:  CALL GETCNT                         ;GET COUNT
 63 017142  012737  000040  002172          MOV #' ,ERRCHR              ;STORE SPACE CHARACTER
 64 017150          CON.S1: PRINTX #ERRONE,#ERRCHR                      ;PRINT THE SPACE
017150  012746  002172                                                  MOV     #ERRCHR,-(SP)
017154  012746  005316                                                  MOV     #ERRONE,-(SP)
017160  012746  000002                                                  MOV     #2,-(SP)
017164  010600                                                          MOV     SP,R0
017166  104415                                                          TRAP    C$PNTX
017170  062706  000006                                                  ADD     #6,SP
 65 017174  005301                  DEC R1                              ;COUNT THE SPACES
 66 017176  001364                  BNE CON.S1
 67 017200  000207                  RETURN                              ;NOW GET NEXT CONTROL CHARACTER
 68
 69                         ;ERROR ROUTINE DISPATCH TABLE
 70
 71 017202  017250  ERRRTB: .WORD CALR1                                 ;CALL ALTERNATE PRINT STRING
 72 017204  017276          .WORD CALR2                                 ;PRINT AN SDI DIAGNOSE RESPONSE
 73         000002          ERRRSZ=<.-ERRRTB>/2
 74
 75 017206  002534  TNAMES: .WORD T1NAME
 76 017210  002556          .WORD T2NAME
 77 017212  002574          .WORD T3NAME
 78 017214  002612          .WORD T4NAME
 79
 80                         ;BUILD TWO TABLES
 81                         ;       FIRST CONTAINING CONTROL CHARACTERS
 82                         ;       SECOND CONTAINING ROUTINE ADDRESSES
 83
 84                         .MACRO BUILD
 85                                 ENTRY '',CON.QU
```

```
                                        ENTRY A,CON.A
                                        ENTRY D,CON.D
                                        ENTRY H,CON.H
                                        ENTRY O,CON.O
                                        ENTRY N,CON.N
                                        ENTRY R,CON.R
                                        ENTRY S,CON.S
                                .ENDM

                                ;HERE IS FIRST TABLE

                                .MACRO ENTRY ARG1,ARG2
                                        .LIST
                                        .BYTE ''ARG1
                                        .NLIST
                                .ENDM

017216                          ERRC:   BUILD
017216  042                             .BYTE '''
017217  101                             .BYTE 'A
017220  104                             .BYTE 'D
017221  110                             .BYTE 'H
017222  117                             .BYTE 'O
017223  116                             .BYTE 'N
017224  122                             .BYTE 'R
017225  123                             .BYTE 'S
017226  000                             .BYTE 0                 ;FOLLOW WITH A NULL BYTE
                                        .EVEN

                                ;HERE IS SECOND TABLE

                                .MACRO ENTRY ARG1,ARG2
                                        .LIST
                                        .WORD ARG2
                                        .NLIST
                                .ENDM

017230                          ERRD:   BUILD
017230  016662                          .WORD CON.QU
017232  016726                          .WORD CON.A
017234  017000                          .WORD CON.D
017236  017012                          .WORD CON.H
017240  017024                          .WORD CON.O
017242  017036                          .WORD CON.N
017244  017070                          .WORD CON.R
017246  017136                          .WORD CON.S
```

GLOBAL SUBROUTINES SECTION

```
                                ;PRE-PROGRAMMED ROUTINE 1
                                ;CALL ALTERNATE PRINT STRING
                                
017250                          CALR1:  PUSH R2                 ;SAVE CURRENT STRING POINTER
017250  010246                                  MOV R2,-(SP)
017252  012402                          MOV (R4)+,R2            ;GET NEW STRING POINTER
017254  006302                          ASL R2                  ;DOUBLE FOR WORD COUNT
017256  063702  002212                  ADD DMPROG,R2           ;ADD START OF STRING STORAGE
017262  067702  162724                  ADD @DMPROG,R2          ;ADD SIZE OF MAIN PROGRAM
017266  004737  016602                  CALL OSTRNG             ;OUTPUT USING THIS STRING
017272                                  POP R2                  ;GET OLD POINTER BACK
017272  012602                                  MOV (SP)+,R2
017274  000207                          RETURN                  ;NOW CONTINUE THE OLD STRING
```

```
                                ;PRE-PROGRAMMED ROUTINE 2
                                ;PRINT AN SDI DIAGNOSE RESPONSE

017276                          CALR2:  PUSH R2                 ;SAVE CURRENT STRING POINTER
017276  010246                                  MOV R2,-(SP)
017300  012402                          MOV (R4)+,R2           ;GET COUNTS
017302                          PUSH R2                         ;SAVE COUNTS
017302  010246                                  MOV R2,-(SP)
017304  000302                          SWAB R2                 ;GET BINARY COUNT
017306  042702  177400                 BIC #177400,R2
017312  001420                          BEQ 2$                  ;BYPASS BINARY IF COUNT IS ZERO
017314  012700  000020         1$:     MOV #16.,R0             ;RADIX IS HEX
017320  012701  000040                 MOV #32.,R1             ;32 BIT NUMBERS
017324  004737  017570                 CALL PNTNUS             ;PRINT THE NUMBER
017330                          PRINTX #ERRNL                   ;GO TO NEW LINE
017330  012746  005321                                  MOV     #ERRNL,-(SP)
017334  012746  000001                                  MOV     #1,-(SP)
017340  010600                                          MOV     SP,R0
017342  104415                                          TRAP    C$PNTX
017344  062706  000004                                  ADD     #4,SP
017350  005302                          DEC R2
017352  001360                          BNE 1$
017354                          2$:     POP R1                  ;GET ASCII COUNT
017354  012601                                  MOV (SP)+,R1
017356  042701  177400                 BIC #177400,R1
017362  001412                          BEQ 3$                  ;BYPASS IS COUNT IS ZERO
017364  004737  016732                 CALL CON.A1             ;PRINT THE ASCII
017370                          PRINTX #ERRNL                   ;GO TO NEW LINE
017370  012746  005321                                  MOV     #ERRNL,-(SP)
017374  012746  000001                                  MOV     #1,-(SP)
017400  010600                                          MOV     SP,R0
017402  104415                                          TRAP    C$PNTX
017404  062706  000004                                  ADD     #4,SP
017410                          3$:     POP R2                  ;RESTORE STRING POINTER
017410  012602                                  MOV (SP)+,R2
017412  000207                          RETURN
```

GLOBAL SUBROUTINES SECTION

```
                                        ;GTDRVT
                                        ;
                                        ;GET DRIVE TABLE POINTER
                                        ;
                                        ;INPUTS:
                                        ;       R5 - CONTROLLER TABLE ADDRESS
                                        ;       R1 - DRIVE NUMBER
                                        ;OUTPUTS:
                                        ;       R4 - DRIVE TABLE ADDRESS
                                        ;       L$LUN - LOADED WITH UNIT NUMBER OF DRIVE
                                        ;       Z CLEAR IF DRIVE TABLE NOT FOUND AFTER ERROR PRINTED

017414                                  GTDRVT: PUSH R2
017414  010246                                          MOV R2,-(SP)
017416  010504                                  MOV R5,R4               ;GET CONTROLLER TABLE ADRESS
017420  062704  000020                          ADD #C.DRO,R4           ;ADD OFFSET TO DRIVE TABLE ADDRESS
017424  012702  000010                          MOV #8.,R2              ;GET COUNT OF DRIVES
017430  005714                          1$:     TST (R4)                ;CHECK IF AN ADDRESS HERE
017432  001406                                  BEQ 3$
017434  027401  000000                          CMP @(R4),R1            ;COMPARE DRIVE NUMBERS
017440  001412                                  BEQ 10$                 ;BRANCH IF A MATCH
017442  005724                          2$:     TST (R4)+               ;BUMP ADDRESS
017444  005302                                  DEC R2
017446  001370                                  BNE 1$                  ;LOOK AT ALL OF THEM
017450                                  3$:     ERRDF 80,GTDRV1         ;UNIT NUMBER NOT FOUND
017450  104455                                                                  TRAP    C$ERDF
017452  000120                                                                  .WORD   80
017454  004547                                                                  .WORD   GTDRV1
017456  000000                                                                  .WORD   0
017460                                          POP R2
017460  012602                                          MOV (SP)+,R2
017462  000244                                  CLZ                     ;CLEAR Z AS ERROR FLAG
017464  000207                                  RETURN
017466  011404                          10$:    MOV (R4),R4             ;GET ADDRESS OF TABLE
017470  016437  000002  002074                  MOV D.UNIT(R4),L$LUN    ;GET UNIT NUMBER
017476                                          POP R2
017476  012602                                          MOV (SP)+,R2
017500  000264                                  SEZ                     ;SET Z FLAG
017502  000207                                  RETURN
```

```
                                        ;GETCNT
                                        ;
                                        ;GET COUNT IN NEXT CHARACTERS OF STRING POINTED TO BY R2.
                                        ;NUMBER WILL BE IN DECIMAL. IF NO NUMBER, RETURN A
                                        ;DEFAULT OF 1.
                                        ;
                                        ;INPUTS:
                                        ;       R2 - POINTER TO ASCII STRING
                                        ;OUTPUTS:
                                        ;       R1 - NUMBER READ OR A ONE
                                        ;       R2 - POINTING TO CHARACTER AFTER NUMBER

017504                                  GETCNT: PUSH R0
017504  010046                                         MOV R0,-(SP)
017506  005001                                 CLR R1                  ;START WITH ZERO COUNT
017510  121227  000060                  GETCNX: CMPB (R2),#'0           ;CHECK IF CHARACTER A DIGIT
017514  103415                                 BLO GETCDN              ;BRANCH IF LOWER THAN ZERO
017516  121227  000071                         CMPB (R2),#'9
017522  101012                                 BHI GETCDN              ;BRANCH IF HIGHER THAN NINE
017524  006301                                 ASL R1                  ;MULTIPLY NUMBER BY 10
017526  010100                                 MOV R1,R0               ; SAVE 2N
017530  006301                                 ASL R1                  ; COMPUTE 4N
017532  006301                                 ASL R1                  ; COMPUTE 8N
017534  060001                                 ADD R0,R1               ; 8N + 2N = 10N
017536  112200                                 MOVB (R2)+,R0           ;GET DIGIT FROM STING
017540  162700  000060                         SUB #'0,R0              ;GET RID OF ASCII
017544  060001                                 ADD R0,R1               ;ADD TO NUMBER
017546  000760                                 BR GETCNX               ;GO TO NEXT CHARACTER
017550  005701                          GETCDN: TST R1                  ;CHECK IF NUMBER IS ZERO
017552  001001                                 BNE GETCXX              ;IF ZERO, CHANGE
017554  005201                                 INC R1                  ; TO DEFAULT OF ONE
017556                                  GETCXX: POP R0
017556  012600                                         MOV (SP)+,R0
017560  000207                                 RETURN
```

```
                                ;PNTNUM
                                ;
                                ;PRINT A NUMBER
                                ;
                                ;INPUTS:
                                ;       R1 - RADIX OF NUMBER
                                ;       R2 - ASCII STRING TO COUNT OF BITS IN NUMBER
                                ;       R4 - POINTER TO NUMBER (LOW WORD)
                                ;OUTPUTS:
                                ;       NUMBER IS PRINTED. LEADING ZEROS ARE PRINTED EXCEPT FOR
                                ;            DECIMAL NUMBERS.
                                ;       R0 - CONTENTS DESTROYED
                                ;
017562  010100          PNTNUM: MOV R1,R0                       ;SAVE RADIX
017564  004737  017504          CALL GETCNT                     ;GET COUNT OF BITS
017570                  PNTNUS: PUSH <R2,R3,R5>
017570  010246                          MOV R2,-(SP)
017572  010346                          MOV R3,-(SP)
017574  010546                          MOV R5,-(SP)
017576  012403                  MOV (R4)+,R3                    ;GET ONE PARAMETER WORD
017600  005005                  CLR R5                          ;CLEAR STORAGE FOR OTHER
017602  020127  000020          CMP R1,#16.                     ;MORE THAN 16 BITS IN NUMBER?
017606  003401                  BLE 1$
017610  012405                  MOV (R4)+,R5                    ;YES, GET SECOND PARAMETER WORD
017612                  1$:     PUSH R4
017612  010446                          MOV R4,-(SP)
017614  010504                  MOV R5,R4                       ;PUT HIGH WORD IN R4
017616  012702  000020          MOV #16.,R2                     ;COMPUTE BITS NOT WANTED
017622  160102                  SUB R1,R2                       ;BY SUBTRACTING BITS TO USE
017624  002002                  BGE 2$                          ;FROM 16.
017626  062702  000020          ADD #16.,R2                     ;IF NEGATIVE, ADD 16 FOR FIRST WORD
017632  001414          2$:     BEQ 6$                          ;IF ZERO, NO BITS NEED BE CLEARED
017634  012705  100000          MOV #BIT15,R5                   ;START MASK WITH SIGN BIT SET
017640  005302          3$:     DEC R2                          ;COUNT BITS IN MASK
017642  001402                  BEQ 4$
017644  006205                  ASR R5                          ;SHIFT MORE BITS TO RIGHT
017646  000774                  BR 3$
017650  020127  000020          4$:     CMP R1,#16.             ;MORE THAN 16 BITS IN NUMBER?
017654  003402                  BLE 5$
017656  040504                  BIC R5,R4                       ;YES, CLEAR IN HIGH WORD
017660  000401                  BR 6$
017662  040503          5$:     BIC R5,R3                       ;NO, CLEAR IN LOW WORD
017664  004737  020070          6$:     CALL DIVIDE             ;DIVIDE BY RADIX IN R0
017670                  PUSH R5                                 ;PUSH REMAINDER ON STACK
017670  010546                          MOV R5,-(SP)
017672  005202                  INC R2                          ;COUNT DIGITS ON STACK
017674  005703                  TST R3                          ;CHECK IF QUOTIENT IS ZERO
017676  001372                  BNE 6$
017700  005704                  TST R4
017702  001370                  BNE 6$
017704  020027  000012          CMP R0,#10.                     ;IF RADIX IS DECIMAL
017710  001434                  BEQ 10$                         ; JUST GO PRINT DIGITS ON STACK
017712  010103                  MOV R1,R3                       ;OTHERWISE COMPUTE NUMBER OF LEADING ZEROS
017714  162700  000014          SUB #12.,R0                     ;DIVIDEND IS BITS IN NUMBER
017720  003002                  BGT 7$                          ;DIVISOR IS BITS PER DIGIT PRINTED
017722  012700  000003          MOV #3,R0                       ;   (3 OR 4)
017726  004737  020070          7$:     CALL DIVIDE
```

```
     53 017732  005705                        TST R5                          ;IF REMAINDER NOT ZERO
     54 017734  001401                        BEQ 8$                          ;INCREMENT QUOTIENT
     55 017736  005203                        INC R3
     56 017740  160203                8$:     SUB R2,R3                       ;SUBTRACT DIGITS ON STACK
     57 017742  001417                        BEQ 10$                         ;NO LEADING ZEROS IF ZERO
     58 017744  112737  000060  002172        MOVB #'0,ERRCHR                 ;STORE ZERO CHARACTER
     59 017752                        9$:     PRINTX #ERRONE,#ERRCHR          ;PRINT A ZERO
        017752  012746  002172                                                          MOV     #ERRCHR,-(SP)
        017756  012746  005316                                                          MOV     #ERRONE,-(SP)
        017762  012746  000002                                                          MOV     #2,-(SP)
        017766  010600                                                                  MOV     SP,R0
        017770  104415                                                                  TRAP    C$PNTX
        017772  062706  000006                                                          ADD     #6,SP
     60 017776  005303                        DEC R3
     61 020000  001364                        BNE 9$                          ;REPEAT UNTIL COUNT REACHES ZERO
     62
     63 020002                        10$:    POP R5                          ;GET CHACACTER FROM STACK
        020002  012605                                  MOV (SP)+,R5
     64 020004  062705  000060                ADD #'0,R5                      ;CNVERT TO ASCII DIGIT
     65 020010  020527  000071                CMP R5,#'9                      ;IF GREATER THAN A 9
     66 020014  003402                        BLE 11$                         ; CONVERT TO A OR HIGHER
     67 020016  062705  000007                ADD #<'A-'9-1>,R5               ; FOR HEX DIGIT
     68 020022  110537  002172        11$:    MOVB R5,ERRCHR                  ;STORE CHARACTER
     69 020026                                PRINTX #ERRONE,#ERRCHR          ;PRINT IT
        020026  012746  002172                                                          MOV     #ERRCHR,-(SP)
        020032  012746  005316                                                          MOV     #ERRONE,-(SP)
        020036  012746  000002                                                          MOV     #2,-(SP)
        020042  010600                                                                  MOV     SP,R0
        020044  104415                                                                  TRAP    C$PNTX
        020046  062706  000006                                                          ADD     #6,SP
     70 020052  005302                DEC R2                                  ;REPEAT FOR ALL DIGITS
     71 020054  001352                BNE 10$                                 ; ON STACK
     72 020056                        POP <R4,R5,R3,R2>
        020056  012604                          MOV (SP)+,R4
        020060  012605                          MOV (SP)+,R5
        020062  012603                          MOV (SP)+,R3
        020064  012602                          MOV (SP)+,R2
     73 020066  000207                RETURN
```

```
                                        ;DIVIDE
                                        ;
                                        ;DIVIDE A 32 BIT UNSIGNED NUMBER BY A 16 BIT UNSIGNED NUMBER.
                                        ;REPLACE DIVIDEND WITH QUOTIENT AND RETURN REMAINDER.
                                        ;WILL NOT CHECK FOR DIVIDE BY ZERO.
                                        ;
                                        ;INPUTS:
                                        ;       R3 - LOW 16 BITS OF DIVIDEND
                                        ;       R4 - HIGH 16 BITS OF DIVIDEND
                                        ;       R0 - DIVISOR
                                        ;OUTPUTS:
                                        ;       R3 - LOW 16 BITS OF QUOTIENT
                                        ;       R4 - HIGH 16 BITS OF QUOTIENT
                                        ;       R5 - REMAINDER
                                        ;
020070                                  DIVIDE: PUSH R2
020070  010246                                         MOV R2,-(SP)
020072  012702  000040                          MOV #32.,R2             ;SET UP SHIFT COUNT
020076  005005                                 CLR R5                  ;START WITH ZERO REMAINDER
020100  006303                          1$:    ASL R3                  ;SHIFT LEFT INTO R5
020102  006104                                 ROL R4
020104  006105                                 ROL R5
020106  020005                                 CMP R0,R5               ;WILL DIVISOR GO INTO REMAINDER
020110  101002                                 BHI 2$                  ;ONLY SUBTRACT IF IT WILL
020112  160005                                 SUB R0,R5               ;SUBTRACT DIVISOR
020114  005203                                 INC R3                  ;PUT A ONE INTO QUOTIENT
020116  005302                          2$:    DEC R2                  ;COUNT THE SHIFTS
020120  001367                                 BNE 1$
020122                                         POP R2
020122  012602                                         MOV (SP)+,R2
020124  000207                          RETURN
```

```
                                ;BUILD DEFAULT 28-BIT NUMBER
                                ;
                                ;INPUT:
                                ;       R4 - POINTER TO 2 WORD DEFAULT NUMBER
                                ;OUTPUT:
                                ;       TEMP - ASCIZ STRING REPRESENTING DEFAULT NUMBER
                                ;
020126                          BLD 8:  PUSH <R0,R1,R3,R4,R5>
020126  010046                                  MOV R0,-(SP)
020130  010146                                  MOV R1,-(SP)
020132  010346                                  MOV R3,-(SP)
020134  010446                                  MOV R4,-(SP)
020136  010546                                  MOV R5,-(SP)
020140  011403                          MOV (R4),R3                     ;GET NUMBER
020142  016404  000002                  MOV 2(R4),R4
020146  012700  000012                  MOV #10.,R0                     ;DIVISOR IS 10.
020152  005001                          CLR R1                          ;CLEAR CHARACTER COUNT
020154  004737  020070          1$:     CALL DIVIDE
020160  062705  000060                  ADD #'0,R5                      ;CONVERT REMAINDER TO ASCII CHARACTER
020164                          PUSH R5                                 ;STORE ON STACK
020164  010546                                  MOV R5,-(SP)
020166  005201                          INC R1                          ;COUNT THE CHARACTER
020170  010305                          MOV R3,R5                       ;REPEAT UNTIL QUOTIENT IS ZERO
020172  050405                          BIS R4,R5
020174  001367                          BNE 1$
020176  012700  002260                  MOV #TEMP,R0                    ;GET POINTER TO STRING
020202                          2$:     POP R5                          ;PUT CHARACTERS INTO STRING
020202  012605                                  MOV (SP)+,R5
020204  110520                          MOVB R5,(R0)+
020206  005301                          DEC R1
020210  001374                          BNE 2$
020212  105020                          CLRB (R0)+                      ;END WITH NULL
020214                          POP <R5,R4,R3,R1,R0>
020214  012605                                  MOV (SP)+,R5
020216  012604                                  MOV (SP)+,R4
020220  012603                                  MOV (SP)+,R3
020222  012601                                  MOV (SP)+,R1
020224  012600                                  MOV (SP)+,R0
020226  000207                          RETURN
```

```
                                ;CONVERT ASCIZ STRING TO 28-BIT NUMBER
                                ;
                                ;INPUTS:
                                ;       TEMP - ASCIZ STRING UP TO 9 CHARACTERS LONK
                                ;       R4 - ADDRESS OF TWO WORD STORAGE
                                ;OUTPUTS:
                                ;       IF STRING IS VALID NUMBER
                                ;         TWO WORDS AT R4 LOADED WITH NUMBER
                                ;         R4 POINTING TO WORD AFTER STORAGE
                                ;         CARRY CLEAR
                                ;       IF STRING INVALID
                                ;         ERROR MESSAGE PRINTED
                                ;         CARRY SET
                                ;
020230                          CNV28:  PUSH <R0,R1,R2,R3>
020230  010046                                 MOV R0,-(SP)
020232  010146                                 MOV R1,-(SP)
020234  010246                                 MOV R2,-(SP)
020236  010346                                 MOV R3,-(SP)
020240  005000                          CLR R0                          ;START WITH ZEROS
020242  005001                          CLR R1
020244  012702  002260                 MOV #TEMP,R2                    ;GET ADDRESS OF STRING
020250  112203                  1$:     MOVB (R2)+,R3                   ;GET A DIGIT FROM STRING
020252  001452                          BEQ 3$                          ;IF NULL CHARACTER, ALL DONE
020254  162703  000060                 SUB #'0,R3                      ;SUBTRACT CHARACTER 0
020260  100435                          BMI 2$
020262  022703  000011                 CMP #9.,R3
020266  103432                          BLO 2$
020270  006300                          ASL R0                          ;MULTIPLY BY 2
020272  006101                          ROL R1
020274                                  PUSH <R1,R0>                    ;SAVE N X 2
020274  010146                                 MOV R1,-(SP)
020276  010046                                 MOV R0,-(SP)
020300  006300                          ASL R0                          ;TIMES 2 AGAIN FOR N X 4
020302  006101                          ROL R1
020304  006300                          ASL R0                          ;TIMES 2 AGAIN FOR N X 8
020306  006101                          ROL R1
020310  062600                          ADD (SP)+,R0                    ;ADD N X 2 TO GIVE N X 10
020312  005501                          ADC R1
020314  062601                          ADD (SP)+,R1
020316  060300                          ADD R3,R0                       ;ADD CURRENT DIGIT
020320  005501                          ADC R1
020322  032701  170000                 BIT #170000,R1                  ;CHECK SIZE OF NUMBER
020326  001750                          BEQ 1$                          ;MUST NOT BE MORE THAN 28 BITS
020330                                  PRINTF #INP28A                  ;PRINT PROPER RANGE
020330  012746  010002                                         MOV     #INP28A,-(SP)
020334  012746  000001                                         MOV     #1,-(SP)
020340  010600                                               MOV     SP,R0
020342  104417                                               TRAP    C$PNTF
020344  062706  000004                                       ADD     #4,SP
020350  000261                          SEC                             ;SET CARRY TO ASK AGAIN
020352  000415                          BR 4$
                                ;
020354                          2$:     PRINTF #INP28B                  ;PRINT ILLEGAL CHARACTER
020354  012746  010044                                         MOV     #INP28B,-(SP)
020360  012746  000001                                         MOV     #1,-(SP)
020364  010600                                               MOV     SP,R0
```

GLOBAL SUBROUTINES SECTION

```
        020366  104417                                                  ..              TRAP    C$PNTF
        020370  062706  000004                                                          ADD     #4,SP
    44  020374  000261                          SEC
    45  020376  000403                          BR 4$
    46
    47  020400  010024                  3$:     MOV R0,(R4)+                    ;MOVE NUMBER TO STORAGE AREA
    48  020402  010124                          MOV R1,(R4)+
    49  020404  000241                          CLC                             ;CLEAR CARRY TO INDICATE ALL IS WELL
    50  020406                          4$:     POP <R3,R2,R1,R0>
        020406  012603                                  MOV (SP)+,R3
        020410  012602                                  MOV (SP)+,R2
        020412  012601                                  MOV (SP)+,R1
        020414  012600                                  MOV (SP)+,R0
    51  020416  000207                          RETURN
```

```
      1                                 ;PRINT HEX NUMBERS WITH LEADING SPACE
      2
      3 020420  112737  000040  002172 T2PNTW: MOVB #' ,ERRCHR                 ;PRINT A SPACE
      4 020426                                 PRINTF #ERRONE,#ERRCHR
        020426  012746  002172                                                 MOV     #ERRCHR,-(SP)
        020432  012746  005316                                                 MOV     #ERRONE,-(SP)
        020436  012746  000002                                                 MOV     #2,-(SP)
        020442  010600                                                         MOV     SP,R0
        020444  104417                                                         TRAP    C$PNTF
        020446  062706  000006                                                 ADD     #6,SP
      5 020452                                 PUSH R1
        020452  010146                                 MOV R1,-(SP)
      6 020454  000301                           SWAB R1
      7 020456  004737  020524                   CALL T2PNT                    ;PRINT HIGH TWO DIGITS
      8 020462                                   POP R1
        020462  012601                                 MOV (SP)+,R1
      9 020464  004737  020524                   CALL T2PNT                    ;PRINT LOW TWO DIGITS
     10 020470  000207                           RETURN
     11
     12 020472  112737  000040  002172 T2PNTB: MOVB #' ,ERRCHR                 ;PRINT A SPACE
     13 020500                                 PRINTF #ERRONE,#ERRCHR
        020500  012746  002172                                                 MOV     #ERRCHR,-(SP)
        020504  012746  005316                                                 MOV     #ERRONE,-(SP)
        020510  012746  000002                                                 MOV     #2,-(SP)
        020514  010600                                                         MOV     SP,R0
        020516  104417                                                         TRAP    C$PNTF
        020520  062706  000006                                                 ADD     #6,SP
     14
     15                                 ;PRINT TWO HEX DIGITS FROM NUMBER IN R1
     16
     17 020524                          T2PNT:  PUSH R1                       ;SAVE NUMBER
        020524  010146                                 MOV R1,-(SP)
     18 020526  006001                          ROR R1                        ;SHIFT TO GET HIGH DIGIT
     19 020530  006001                          ROR R1
     20 020532  006001                          ROR R1
     21 020534  006001                          ROR R1
     22 020536  004737  020544                  CALL T2PNTO                   ;PRINT TWO DIGITS
     23 020542                                  POP R1                        ;GET LOW DIGIT AGAIN
        020542  012601                                 MOV (SP)+,R1
     24 020544  042701  177760          T2PNTO: BIC #^C17,R1                  ;CLEAR OTHER BITS
     25 020550  062701  000060                  ADD #'0,R1                    ;CONVERT TO ASCII CHARACTER
     26 020554  020127  000071                  CMP R1,#'9                    ;IF GREATER THAN A 9
     27 020560  003402                          BLE T2PNTD                    ;  CONVERT TO A OR HIGHER
     28 020562  062701  000007                  ADD #<'A-'9-1>,R1             ;  FOR HEX DIGIT
     29 020566  110137  002172          T2PNTD: MOVB R1,ERRCHR                ;PRINT THE DIGIT
     30 020572                                  PRINTF #ERRONE,#ERRCHR
        020572  012746  002172                                                 MOV     #ERRCHR,-(SP)
        020576  012746  005316                                                 MOV     #ERRONE,-(SP)
        020602  012746  000002                                                 MOV     #2,-(SP)
        020606  010600                                                         MOV     SP,R0
        020610  104417                                                         TRAP    C$PNTF
        020612  062706  000006                                                 ADD     #6,SP
     31 020616  000207                          RETURN
```

GLOBAL SUBROUTINES SECTION

```
                                        ;T2GNUM
                                        ;
                                        ;GET A HEX DIGIT FROM AN ASCII INPUT STRING
                                        ;
                                        ;INPUTS:
                                        ;       R1 - STRING POINTER
                                        ;OUTPUTS:
                                        ;       R4 - NUMBER
                                        ;       R1 - UPDATED STRING TO CHARACTER AFTER NUMBER
                                        ;       R0 - COUNT OF DIGITS (0 IF END OF LINE FOUND)

020620  005000                  T2GNUM: CLR R0                          ;CLEAR DIGIT COUNT
020622  105711                          TSTB (R1)                       ;CHECK IF END OF LINE
020624  001442                          BEQ T2GNX                       ;REPORT NULL CHARACTER FOUND
020626  121127  000040                  CMPB (R1),#'                    ;CHECK IF A SPACE
020632  001002                          BNE T2GND1                      ;IF SO, IGNORE IT
020634  005201                          INC R1
020636  000770                          BR T2GNUM
020640  005004                  T2GND1: CLR R4                          ;CLEAR NUMBER STORAGE
020642                          T2GND2: PUSH R2                         ;SAVE REGISTER
020642  010246                                  MOV R2,-(SP)
020644  112102                          MOVB (R1)+,R2                   ;GET CHARACTER
020646  162702  000060                  SUB #'0,R2                      ;CONVERT TO HEX DIGIT
020652  100431                          BMI T2GNE
020654  020227  000011                  CMP R2,#9.
020660  003410                          BLE T2GND3
020662  020227  000021                  CMP R2,#<'A-'0>
020666  103423                          BLO T2GNE
020670  020227  000026                  CMP R2,#<'F-'0>
020674  101020                          BHI T2GNE
020676  162702  000007                  SUB #<'A-'9-1>,R2
020702  006304                  T2GND3: ASL R4
020704  006304                          ASL R4
020706  006304                          ASL R4
020710  006304                          ASL R4
020712  050204                          BIS R2,R4
020714  005200                          INC R0
020716                                  POP R2
020716  012602                                  MOV (SP)+,R2
020720  105711                          TSTB (R1)
020722  001403                          BEQ T2GNX
020724  121127  000040                  CMPB (R1),#'
020730  001344                          BNE T2GND2
020732  005700                  T2GNX:  TST R0
020734  000207                          RETURN

020736                          T2GNE:  POP R2                          ;CLEAN UP THE STACK
020736  012602                                  MOV (SP)+,R2
020740                                  POP R0
020740  012600                                  MOV (SP)+,R0
020742  000137  015126                  JMP T2CMDE
```

GLOBAL SUBROUTINES SECTION

```
                                ;PNTERR
                                ;
                                ;PRINT ERROR MESSAGE FROM DM PROGRAM REQUEST 11 OR 12.
                                ;
                                ;INPUTS:
                                ;       R5 - CONTROLLER TABLE ADDRESS
                                ;       R4 - MESSAGE DATA ADDRESS
                                ;OUTPUTS:
                                ;       NONE

020746                          PNTERR: PUSH <R0,R1,R2,R3>
020746  010046                                  MOV R0,-(SP)
020750  010146                                  MOV R1,-(SP)
020752  010246                                  MOV R2,-(SP)
020754  010346                                  MOV R3,-(SP)
020756  005764  000004                 TST 4(R4)                       ;GET DRIVE NUMBER
020762  002004                          BGE 1$                          ;CHECK IF BIT 15 SET
020764  016537  000002  C02074         MOV C.UNIT(R5),L$LUN            ;IF SO, GET UNIT FROM CONTROLLER TABLE
020772  000406                          BR 2$
020774                          1$:     PUSH R4                         ;SAVE DATA ADDRESS
020774  010446                                  MOV R4,-(SP)
020776  016401  000004                 MOV 4(R4),R1                    ;GET DRIVE NUMBER
021002  004737  017414                 CALL GTDRVT                     ;GET DRIVE TABLE ADDRESS
021006                                  POP R4                          ;RESTORE DATA ADDRESS
021006  012604                                  MOV (SP)+,R4
021010  012702  002162          2$:     MOV #ERRTYP,R2                  ;GET POINTER TO ERROR TABLE
021014  116422  000003                 MOVB 3(R4),(R2)+                ;GET ERROR TYPE
021020  105022                          CLRB (R2)+                      ;CLEAR HIGH BYTE
021022  116422  000002                 MOVB 2(R4),(R2)+                ;GET ERROR NUMBER
021026  105022                          CLRB (R2)+                      ;CLEAR HIGH BYTE
021030  005022                          CLR (R2)+                       ;CLEAR MESSAGE POINTER
021032  012712  021052                 MOV #ERRRTN,(R2)                ;GET ROUTINE NUMBER
021036                                  ERROR                           ;PRINT THE ERROR MESSAGE
021036  104460                                                                  TRAP    C$ERROR
021040                                  POP <R3,R2,R1,R0>
021040  012603                                  MOV (SP)+,R3
021042  012602                                  MOV (SP)+,R2
021044  012601                                  MOV (SP)+,R1
021046  012600                                  MOV (SP)+,R0
021050  000207                          RETURN

021052                          BGNMSG ERRRTN                           ;ERROR REPORT ROUTINE
021052                                                                          ERRRTN::
021052  013702  002226                 MOV TNUM,R2                     ;GET TEST NUMBER
021056  006302                          ASL R2                          ;DOUBLE
021060                                  PRINTB #ERRMB,TNAMES-2(R2),(R4),C.UADR(R5)      ;PRINT BASIC MESSAGE LINE WITH DM PC
021060  016546  000000                                                          MOV     C.UADR(R5),-(SP)
021064  011446                                                                  MOV     (R4),-(SP)
021066  016246  017204                                                          MOV     TNAMES-2(R2),-(SP)
021072  012746  006216                                                          MOV     #ERRMB,-(SP)
021076  012746  000004                                                          MOV     #4,-(SP)
021102  010600                                                                  MOV     SP,R0
021104  104414                                                                  TRAP    C$PNTB
021106  062706  000012                                                          ADD     #12,SP
021112  005764  000004                 TST 4(R4)                       ;CHECK IF DRIVE NUMBER GIVEN
021116  100412                          BMI 1$                          ;BRANCH IF NOT
021120                                  PRINTB #ERRMBD,4(R4)            ;PRINT DRIVE NUMBER
```

```
        021120  016446  000004                                                  MOV     4(R4),-(SP)
        021124  012746  006260                                                  MOV     #ERRMBD,-(SP)
        021130  012746  000002                                                  MOV     #2,-(SP)
        021134  010600                                                          MOV     SP,R0
        021136  104414                                                          TRAP    C$PNTB
        021140  062706  000006                                                  ADD     #6,SP
    38  021144  004737  024436          1$:    CALL RNTIME                     ;PRINT RUNTIME
    39  021150  062704  000006                 ADD #6,R4                       ;INCREASE R4 TO POINT TO MESSAGE POINTER
    40  021154  012402                         MOV (R4)+,R2                    ;GET MESSAGE POINTER
    41  021156  006302                         ASL R2                          ;DOUBLE TO MAKE BYTE OFFSET
    42  021160  063702  002212                 ADD DMPROG,R2                   ;ADD TO START OF MESSAGE STRINGS
    43  021164  067702  161022                 ADD @DMPROG,R2                  ;ADD SIZE OF MAIN PROGRAM
    44  021170  105712                         TSTB (R2)                       ;CHECK FIRST BYTE
    45  021172  001001                         BNE NCON                        ;IF ZERO
    46  021174  005202                         INC R2                          ; INCREMENT TO NEXT BYTE
    47  021176  004737  016602          NCON:  CALL OSTRNG                     ;OUTPUT ACCORDING TO STRING
    48  021202                                 ENDMSG
        021202                                                          L10020:
        021202  104423                                                          TRAP    C$MSG
    49
```

GLOBAL SUBROUTINES SECTION

```
      1                                 ;LOADDM
      2                                 ;
      3                                 ;LOAD AND START A DM PROGRAM INTO A CONTROLLER
      4                                 ;
      5                                 ;INPUTS:
      6                                 ;       R5 - CONTROLLER TABLE ADDRESS
      7                                 ;IMPLICIT INPUTS:
      8                                 ;       DMPROG - POINTER TO START OF DM PROGRAM IN MEMORY
      9                                 ;OUTPUTS:
     10                                 ;       IF LOAD SUCCEEDS - Z CLEAR
     11                                 ;                         CONTROLLER TABLE MARKED LOADED
     12                                 ;       IF ERROR - Z SET
     13
     14 021204                          LOADDM:
     28 021204  016504  000004                  MOV C.VEC(R5),R4                ;GET VECTOR OF UDA
     29 021210                                  AND CT.VEC,R4
        021210  042704  177000                          BIC #^C<CT.VEC>,R4
     30 021214  010501                          MOV R5,R1                       ;GET INTERRUPT SERVICE LINK
     31 021216  062701  000010                  ADD #C.JSR,R1
     32 021222                                  SETVEC R4,R1,#PRI07             ;SET UP INTERRUPT VECTOR
        021222  012746  000340                                                          MOV     #PRI07,-(SP)
        021226  010146                                                                  MOV     R1,-(SP)
        021230  010446                                                                  MOV     R4,-(SP)
        021232  012746  000003                                                          MOV     #3,-(SP)
        021236  104437                                                                  TRAP    C$SVEC
        021240  062706  000010                                                          ADD     #10,SP
     33                                                                 ;INITIALIZE UDA WITH SMALLEST
     34 021244  006204                          ASR R4                          ;POSITION VECTOR FOR UDA
     35 021246  006204                          ASR R4
     36 021250  004737  022742                  CALL UDAINT                     ; RING BUFFER AND INTERRUPTS ENABLED
     37 021254  001561                          BEQ LOADER                      ;BRANCH IF AN ERROR
     38 021256  004737  012746                  CALL HCOMM                      ;ALLOCATE SPACE FOR HOST COMM AREA
     39
     40 021262  023727  002226  000001          CMP TNUM,#1                     ;IF TEST NUMBER 1
     41 021270  001441                          BEQ LOADT1                      ;  DO SPECIAL LOAD
     42 021272  017701  160714                  MOV @DMPROG,R1                  ;GET SIZE OF PROGRAM
     43 021276  012700  000002          LOADB:  MOV #OP.ESP,R0                  ;BUILD EXECUTE SUPPLIED PROGRAM COMMAND PACKET
     44 021302  004737  021624                  CALL BLDCMD
     45 021306  013764  002212  000124          MOV DMPROG,HC.CPK+P.UADR(R4)    ;LOAD MAIN PROGRAM ADDRESS
     46 021314  010164  000120                  MOV R1,HC.CPK+P.BCNT(R4)        ;  AND SIZE
     47 021320  013764  002212  000140          MOV DMPROG,HC.CPK+P.OVRL(R4)    ;LOAD OVERLAY ADDRESS
     48 021326  067764  160660  000140          ADD @DMPROG,HC.CPK+P.OVRL(R4)
     52 021334  004737  021710                  CALL SNDCMD                     ;SEND COMMAND TO UDA
     53 021340  001527                          BEQ LOADER                      ;GET OUT IF ERROR
     54 021342  004737  022050                  CALL WAITMS                     ;WAIT FOR MESSAGE RESPONSE
     55 021346  032764  000037  000032          BIT #ST.MSK,HC.MPK+P.STS(R4)    ;CHECK FOR ERRORS
     56 021354  001115                          BNE LOADE1
     57 021356  042765  000024  000014          BIC #CT.CMD+CT.REQ,C.FLG(R5)    ;CLEAR COMMAND OUTSTANDING FLAG
     58 021364  052765  000002  000014          BIS #CT.RN,C.FLG(R5)            ;SET DM PROGRAM RUNNING FLAG
     59 021372  000207                          RETURN
```

```
                                ;LOAD DM PROGRAM FROM MEMORY SPACE TESTED DURING
                                ;INITIALIZATION IN TEST 1
                                
021374  017704  160612    LOADT1: MOV @DMPROG,R4          ;GET SIZE OF DM PROGRAM IN BYTES
021400  162704  000040            SUB #DMMAIN,R4
021404  013700  002212            MOV DMPROG,R0           ;GET ADDRESS OF DM PROGRAM
021410  062700  000040            ADD #DMMAIN,R0
021414  005001                    CLR R1                  ;START WITH OFFSET OF ZERO
                                
021416  012703  000170    LT1L1:  MOV #<HC.BSZ*2>,R3      ;GET SIZE OF BOTH BUFFERS
021422  020403                    CMP R4,R3               ;IF FEWER BYTES REMAINING IN PROGRAM
021424  103001                    BHIS LT11
021426  010403                    MOV R4,R3               ;USE ACTUAL BYTE COUNT
021430                  LT11:   PUSH R3                 ;SAVE THE BYTE COUNT
        010346                          MOV R3,-(SP)
021432  013702  002174            MOV FFREE,R2            ;GET ADDRESS OF BUFFER
021436  162702  000170            SUB #<HC.BSZ*2>,R2
021442                          PUSH R2                 ;SAVE BUFFER ADDRESS
        010246                          MOV R2,-(SP)
021444  012022          LT1L2:  MOV (R0)+,(R2)+         ;MOVE DATA TO BUFFER
021446  162703  000002            SUB #2,R3               ;COUNT BYTES
021452  001374                    BNE LT1L2
021454                          POP R2                  ;RESTORE BUFFER ADDRESS
        012602                          MOV (SP)+,R2
021456                          POP R3                  ;RESTORE BYTE COUNT
        012603                          MOV (SP)+,R3
021460  004737  021506            CALL LOAD               ;LOAD INTO UDA
021464  001455                    BEQ LOADER              ;IF ERROR, GET OUT NOW
021466  006203                    ASR R3                  ;CONVERT BYTES TO WORDS
021470  060301                    ADD R3,R1               ;INCREASE OFFSET FOR NEXT BUFFER
021472  006303                    ASL R3                  ;CONVERT WORDS TO BYTES
021474  160304                    SUB R3,R4               ;REDUCE REMAINING BYTE COUNT
021476  001347                    BNE LT1L1               ;GET NEXT BUFFER
021500  012701  000040            MOV #DMMAIN,R1          ;GET A BYTE COUNT OF HEADER ONLY
021504  000674                    BR LOADB                ;NOW START
```

```
                                        ;LOAD
                                        ;
                                        ;ISSUE DOWNLINE LOAD COMMAND TO UDA. CHECK THAT LOAD
                                        ;HAPPENS WITHOUT ERROR.
                                        ;
                                        ;INPUTS:
                                        ;       R1 - OFFSET FOR DM PROGRAM
                                        ;       R2 - ADDRESS OF BUFFER CONTAINING PROGRAM
                                        ;       R3 - SIZE OF BUFFER IN BYTES
                                        ;       R5 - CONTROLLER TABLE ADDRESS
                                        ;OUTPUTS:
                                        ;       Z CLEAR IF NO ERROR
                                        ;       Z SET IF ERROR AND ERROR REPORTED
                                        ;
021506                                  LOAD:   PUSH <R0,R3,R4>
021506  010046                                          MOV R0,-(SP)
021510  010346                                          MOV R3,-(SP)
021512  010446                                          MOV R4,-(SP)
021514  012700  000031                    MOV #OP.MWR,R0                  ;GET DOWNLINE LOAD COMMAND
021520  004737  021624                    CALL BLDCMD                     ;BUILD COMMAND PACKET
021524  010264  000124                    MOV R2,HC.CPK+P.UADR(R4)        ;STUFF IN BUFFER ADDRESS
021530  010364  000120                    MOV R3,HC.CPK+P.BCNT(R4)        ;STUFF IN BYTE COUNT
021534  010164  000144                    MOV R1,HC.CPK+P.RGOF(R4)        ;STUFF IN OFFSET
021540  012764  000001  000140            MOV #1,HC.CPK+P.RGID(R4)        ;STUFF IN REGION ID 1
021546  004737  021710                    CALL SNDCMD                     ;SEND COMMAND TO UDA
021552  001422                          BEQ LOADER
021554  004737  022050                    CALL WAITMS                     ;WAIT FOR MESSAGE RESPONSE
021560  032764  000037  000032          BIT #ST.MSK,HC.MPK+P.STS(R4)    ;LOOK FOR ANY ERROR
021566  001010                          BNE LOADE1
021570  042765  000004  000014          BIC #CT.CMD,C.FLG(R5)           ;CLEAR COMMAND ISSUED
021576                                  POP <R4,R3,R0>
021576  012604                                  MOV (SP)+,R4
021600  012603                                  MOV (SP)+,R3
021602  012600                                  MOV (SP)+,R0
021604  000244                          CLZ                             ;CLEAR Z TO INDICATE NO ERROR
021606  000207                          RETURN
```

```
                                ;UDA FAILED TO DOWNLINE LOAD DM PROGRAM
                                
021610                          LOADE1: ERRHRD 20,LOADM1
021610  104456                                                    TRAP    C$ERHRD
021612  000024                                                    .WORD   20
021614  003206                                                    .WORD   LOADM1
021616  000000                                                    .WORD   0
021620  000264                  LOADER: SEZ                     ;SET Z TO INDICATE ERROR OCCURRED
021622  000207                          RETURN
```

```
                                ;BLDCMD
                                ;
                                ;BUILD A COMMAND IN COMMAND PACKET
                                ;
                                ;INPUTS:
                                ;       R5 - CONTROLLER TABLE ADDRESS
                                ;       R0 - COMMAND CODE
                                ;OUTPUTS:
                                ;       R4 - ADDRESS OF HOST COMM AREA
                                ;       COMMAND PACKET CONTAINING REF NUMBER AND OPCODE. ALL OTHER FIELDS CLEARED.
                                ;       CMD REFERRENCE NUMBER IN CONTROLLER TABLE INCREMENTED AND RESULT
                                ;       IN COMMAND PACKET.
                                ;       R0 - CONTENTS DESTROYED
                                ;
021624                          BLDCMD: PUSH <R1,R0>
021624  010146                                  MOV R1,-(SP)
021626  010046                                  MOV R0,-(SP)
021630  016504  000016                MOV C.RING(R5),R4               ;GET ADDRESS OF HOST COMM AREA
021634  010400                        MOV R4,R0                       ;COPY TO R0
021636  062700  000100                ADD #HC.CEV,R0                  ;COMPUTE ADDRESS OF COMMAND ENVELOPE
021642  012720  000060                MOV #HC.PSZ,(R0)+               ;LOAD PACKET LENGTH
021646  012701  001000                MOV #DUP,R1                     ;LOAD DIAG CIRCUIT IDENTIFIER
021652  022716  000031                CMP #OP.MWR,(SP)                ;IF CODE IS MAINTENANCE WRITE
021656  001002                        BNE BLDC0                       ;  GET OTHER CIRCUIT IDENTIFIER
021660  012701  177777                MOV #DIAG,R1
021664  010120                  BLDC0:  MOV R1,(R0)+                    ;PUT IDENTIFIER INTO PACKET
021666  012701  000030                MOV #<HC.PSZ>/2,R1              ;GET WORDS TO CLEAR
021672  005020                  BLDC1:  CLR (R0)+                       ;CLEAR PACKET
021674  005301                        DEC R1
021676  001375                        BNE BLDC1
021700                          POP HC.CPK+P.OPCD(R4)                   ;PUT OPCODE IN PACKET
021700  012664  000114                        MOV (SP)+,HC.CPK+P.OPCD(R4)
021704                          POP R1                                  ;RESTORE R1
021704  012601                                MOV (SP)+,R1
021706  000207                  RETURN
```

```
                                        ;SNDCMD
                                        ;
                                        ;SEND A COMMAND TO THE UDA.
                                        ;CLEAR THE RESPONSE PACKET. MARK BOTH PACKETS AVAILABLE TO THE
                                        ;UDA. SET COMMAND ISSUED BIT IN CONTROLLER TABLE AND INITIALIZE
                                        ;TIMEOUT COUNTER.
                                        ;
                                        ;INPUTS:
                                        ;       R5 - CONTROLLER TABLE ADDRESS
                                        ;OUTPUTS:
                                        ;       R4 - ADDRESS OF HOST COMM AREA


021710                                  SNDCMD: PUSH <R0,R1>
021710  010046                                        MOV R0,-(SP)
021712  010146                                        MOV R1,-(SP)
021714  016504  000016                              MOV C.RING(R5),R4               ;LOAD R4 WITH HOST COMM AREA ADDRESS
021720  005265  000044                              INC C.REF(R5)                   ;INCREMENT CMD REFERENCE NUMBER
021724  016564  000044  000104                      MOV C.REF(R5),HC.CPK+P.CRF(R4)  ;PUT IN PACKET
021732  012700  000014                              MOV #HC.MEV,R0                  ;POINT TO MESSAGE ENVELOPE
021736  060400                                      ADD R4,R0
021740  012701  000032                              MOV #<HC.PSZ+HC.ESZ>/2,R1       ;SIZE OF MESSAGE PACKET
021744  005020                          SNDC1:      CLR (R0)+                       ;CLEAR ENTIRE MESSAGE PACKET
021746  005301                                      DEC R1
021750  001375                                      BNE SNDC1
021752  012764  140000  000006                      MOV #RG.OWN+RG.FLG,HC.MCT(R4)   ;MARK MESSAGE PACKET AVAILABLE
021760  012764  100000  000012                      MOV #RG.OWN,HC.CCT(R4)          ;MARK COMMAND TO UDA
021766  005775  000000                              TST @(R5)                       ;TELL UDA COMMAND IS THERE
021772  052765  000004  000014                      BIS #CT.CMD,C.FLG(R5)           ;MARK COMMAND ISSUED
022000                                  POP         <R1,R0>
022000  012601                                      MOV (SP)+,R1
022002  012600                                      MOV (SP)+,R0
022004  000207                          RETURN
```

```
                                        ;CLRBUF
                                        ;
                                        ;CLEAR THE SPECIFIED DATA BUFFER IN THE HOST COMM AREA
                                        ;AND LOAD BUFFER DESCRIPTOR IN COMMAND PACKET TO THE BUFFER
                                        ;
                                        ;INPUTS:
                                        ;       R5 - CONTROLLER TABLE ADDRESS
                                        ;       R4 - ADDRESS OF HOST COMM AREA
                                        ;       R0 - OFFSET INTO HOST COMM AREA TO DATA BUFFER
                                        ;OUTPUTS:
                                        ;       DATA BUFFER CLEARED
                                        ;       COMMAND PACKET POINTING TO BUFFER
                                        ;       BYTE COUNT SET TO SIZE OF BUFFER
                                        ;       R4 - ADDRESS OF DATA BUFFER

022006                                  CLRBUF: PUSH <R0,R1>
022006  010046                                         MOV R0,-(SP)
022010  010146                                         MOV R1,-(SP)
022012  060400                                 ADD R4,R0                      ;ADD START OF HOST COMM AREA TO OFFSET
022014  010064  000124                         MOV R0,HC.CPK+P.UADR(R4)       ;PUT BUFFER ADDRESS IN COMMAND PACKET
022020  012764  000074  000120                 MOV #HC.BSZ,HC.CPK+P.BCNT(R4)  ;PUT SIZE OF BUFFER IN COMMAND PACKET
022026  010004                                 MOV R0,R4                      ;PUT BUFFER ADDRESS IN R4
022030  012701  000036                         MOV #HC.BSZ/2,R1               ;GET SIZE OF BUFFER IN WORDS
022034  005020                          CLRBFL: CLR (R0)+                      ;CLEAR ALL THE WORDS
022036  005301                                 DEC R1
022040  001375                                 BNE CLRBFL
022042                                         POP <R1,R0>
022042  012601                                         MOV (SP)+,R1
022044  012600                                         MOV (SP)+,R0
022046  000207                                 RETURN
```

```
                                ;WAITMS
                                ;
                                ;WAIT FOR UDA TO RESPOND WITH A MESSAGE PACKET
                                ;
                                ;INPUTS:
                                ;       R5 - ADDRESS OF CONTROLLER TABLE
                                ;OUTPUTS:
                                ;       Z CLEAR IF NO ERROR
                                ;       Z SET IF ERROR, MESSAGE PRINTED
                                ;
022050                          WAITMS: PUSH <R0,R1>
022050  010046                                MOV R0,-(SP)
022052  010146                                MOV R1,-(SP)
022054  012700  000036                    MOV #30.,R0             ;SET TIME OUT VALUE OF 30 SECONDS
022060  010501                          MOV R5,R1               ;POINT TO TIME OUT COUNTER
022062  062701  000040                  ADD #C.TO,R1
022066  004737  022660                  CALL SETTO
022072  011500                          MOV (R5),R0             ;GET ADDRESS OF UDAIP REGISTER
022074                          1$:
022074  032765  000010  000014          BIT #CT.MSG,C.FLG(R5)   ;LOOK IF INTERRUPT OCCURRED
022102  001030                          BNE 3$                  ;BRANCH IF SO
022104  016001  000002                  MOV 2(R0),R1            ;LOOK AT UDASA REGISTER
022110  001034                          BNE 4$                  ;BRANCH IF ERROR CODE PRESENT
022112                                  BREAK
022112  104422                                                          TRAP    C$BRK
022114  005737  002352                  TST KW.CSR              ;SEE IF A CLOCK ON SYSTEM
022120  001765                          BEQ 1$
022122  023765  002364  000042          CMP KW.EL+2,C.TOH(R5)   ;CHECK IF TIMEOUT HAS HAPPENED
022130  101003                          BHI 2$
022132  001360                          BNE 1$
022134  023765  002362  000040          CMP KW.EL,C.TO(R5)
022142  103754                          BLO 1$
022144                          2$:     ERRDF 39,0,RSPTOE
022144  104455                                                          TRAP    C$ERDF
022146  000047                                                          .WORD   39
022150  000000                                                          .WORD   0
022152  012636                                                          .WORD   RSPTOE
022154                                  POP <R1,R0>
022154  012601                                  MOV (SP)+,R1
022156  012600                                  MOV (SP)+,R0
022160  000264                          SEZ
022162  000207                          RETURN
022164  042765  000010  000014  3$:     BIC #CT.MSG,C.FLG(R5)   ;CLEAR MESSAGE RECEIVED FLAG
022172                                  POP <R1,R0>
022172  012601                                  MOV (SP)+,R1
022174  012600                                  MOV (SP)+,R0
022176  000244                          CLZ                     ;GIVE NO ERROR RETURN
022200  000207                          RETURN
022202                          4$:     ERRDF 40,0,RSPTME
022202  104455                                                          TRAP    C$ERDF
022204  000050                                                          .WORD   40
022206  000000                                                          .WORD   0
022210  012662                                                          .WORD   RSPTME
022212                                  POP <R1,R0>
022212  012601                                  MOV (SP)+,R1
022214  012600                                  MOV (SP)+,R0
022216  000264                          SEZ
```

```
     47 022220  000207                          RETURN
```

```
                                        ;APRINT
                                        ;
                                        ;CONVERT AN 18 BIT ADDRESS STORED IN TWO WORDS INTO A FORMAT
                                        ;THAT WILL ALLOW PRINTING OF THE 18 BIT NUMBER.
                                        ;
                                        ;INPUTS:
                                        ;       R0 - ADDRESS OF TWO WORD BLOCK CONTAINING ADDRESS.
                                        ;               FIRST WORD CONTAINING LOW 16 BITS.
                                        ;               SECOND WORD CONTAINING HIGH 2 BITS.
                                        ;OUTPUTS:
                                        ;       R1 - HIGH 3 BITS OF ADDRESS
                                        ;       R2 - LOW 15 BITS OF ADDRESS
                                        
022222  016001  000002          APRINT: MOV 2(R0),R1                    ;GET HIGH 2 BITS
022226  006301                          ASL R1                          ;SHIFT LEFT
022230  011002                          MOV (R0),R2                     ;GET LOW 16 BITS
022232  100001                          BPL APRIZ                       ;IF 16TH BIT SET
022234  005201                          INC R1                          ;PLACE IT IN WITH HIGH 2 BITS
022236  000207                  APRIZ:  RETURN
```

```
                                ;INITKT
                                ;
                                ;INITIALIZE THE KT OPTION (IF AVAILABLE) SO THAT IT CAN BE TURNED
                                ;ON AND OFF BY KTREAD AND KTWRITE ROUTINES.
                                ;
                                ;INPUTS:
                                ;       NONE
                                ;OUTPUTS:
                                ;       ALL KT REGISTERS LOADED SO THAT IF THE KT OPTION IS TURNED ON, ALL
                                ;       MEMORY WOULD APPEAR THE SAME AS WHEN TURNED OFF. KT OPTION IS TURNED
                                ;       OFF IF ON WHEN CALLED.
                                ;
                                ;       KTMEM - 0 IF KT EXISTS, -1 IF KT DOES NOT EXIST.

022240                          INITKT: PUSH R0
022240  010046                                MOV R0,-(SP)
022242  005037  002366                  CLR NXMAD               ;CLEAR MEMORY FLAG
022246                                  SETVEC #4,#NXMI,#PRI07  ;TRAP NON-EXISTANT ADDRESSES
022246  012746  000340                                                  MOV     #PRI07,-(SP)
022252  012746  022640                                                  MOV     #NXMI,-(SP)
022256  012746  000004                                                  MOV     #4,-(SP)
022262  012746  000003                                                  MOV     #3,-(SP)
022266  104437                                                          TRAP    C$SVEC
022270  062706  000010                                                  ADD     #10,SP
022274  005037  177572                  CLR SR0                 ;MAKE SURE MEMORY MANAGEMENT IS OFF
022300  012700  077406                  MOV #MM.PLF+MM.ACF,R0   ;FILL PDR'S
022304  010037  172300                  MOV R0,PDR0             ;TO ALLOW READ AND WRITE
022310  010037  172302                  MOV R0,PDR1             ;TO ALL LOCATIONS
022314  010037  172304                  MOV R0,PDR2
022320  010037  172306                  MOV R0,PDR3
022324  010037  172310                  MOV R0,PDR4
022330  010037  172312                  MOV R0,PDR5
022334  010037  172314                  MOV R0,PDR6
022340  010037  172316                  MOV R0,PDR7
022344  005037  172340                  CLR PAR0                ;POINT PAGE ADDRESS REGISTERS
022350  012737  000200  172342          MOV #200,PAR1           ;AT SAME PHYSICAL ADDRESS
022356  012737  000400  172344          MOV #400,PAR2
022364  012737  000600  172345          MOV #600,PAR3
022372  012737  001000  172350          MOV #1000,PAR4
022400  012737  001200  172352          MOV #1200,PAR5
022406  012737  001400  172354          MOV #1400,PAR6
022414  012737  007600  172356          MOV #7600,PAR7
022422                                  CLRVEC #4               ;RELEASE THE VECTOR
022422  012700  000004                                                  MOV     #4,R0
022426  104436                                                          TRAP    C$CVEC
022430  013737  002366  002370          MOV NXMAD,KTMEM         ;STORE KT EXISTS FLAG
022436  005037  002222                  CLR KTBASO              ;CLEAR OFFSET TO REACH BASE
022442  005037  002220                  CLR KTBASA              ;CLEAR BASE ADDRESS
022446                                  POP R0
022446  012600                                  MOV (SP)+,R0
022450  000207                          RETURN
```

```
                                ;KTBASE
                                ;
                                ;SET BASE ADDRESS FOR BUFFER IN MEMORY. KT READ AND KTWRITE CALLS
                                ;WILL THEN REFFRENCE THE BUFFER USING AN OFFSET THAT GETS ADDRESS
                                ;RELATIVE TO BASE.
                                ;
                                ;INPUTS:
                                ;       R0 - ADDRESS OF TWO WORD BLOCK CONTAINING 18 BIT ADDRESS
                                ;OUTPUTS:
                                ;       NONE

                                KTBASE: PUSH R0
022452  010046                                  MOV R0,-(SP)
                                        MOVB (R0)+,KTBASO               ;FIRST BYTE IS OFFSET
                                        MOVB (R0)+,KTBASA               ;SECOND AND THIRD BYTES
                                        MOVB (R0)+,KTBASA+1             ; MAKE THE BASE ADDRESS
                                        POP R0
022470  012600                                  MOV (SP)+,R0
                                        RETURN
```

GLOBAL SUBROUTINES SECTION

```
                                        ;KTREAD
                                        ;
                                        ;READ ONE WORD FROM MEMORY USING KT HARDWARE. WORD IS SPECIFIED BY A BASE
                                        ;ADDRESS AND A BYTE OFFSET. THE BASE ADDRESS MUST BE PREVIOUSLY SPECIFIED
                                        ;BY A KTBASE CALL.
                                        ;
                                        ;INPUTS:
                                        ;       R1 - BYTE OFFSET INTO BUFFER
                                        ;OUTPUTS:
                                        ;       R0 - DATA READ FROM MEMORY LOCATION

022474                                  KTREAD: PUSH R1
022474  010146                                  MOV R1,-(SP)
022476  063701  002222                        ADD KTBASO,R1                   ;COMPUTE TOTAL OFFSET
022502                                  PUSH R1
022502  010146                                  MOV R1,-(SP)
022504  000301                          SWAB R1                         ;GET RID OF LOW BYTE
022506  042701  177400                 BIC #177400,R1
022512  063701  002220                 ADD KTBASA,R1                   ;ADD BASE ADDRESS
022516  006301                          ASL R1                          ;SHIFT TO POSITIN FOR PAR REGISTER
022520  006301                          ASL R1
022522  010137  172354                 MOV R1,PAR6                     ;STORE IN PAR REGISTER
022526                                  POP R1
022526  012601                                  MOV (SP)+,R1
022530  042701  177400                 BIC #177400,R1                  ;CLEAR HIGH BYTE
022534  012737  000001  177572         MOV #MM.EN,SR0                  ;TURN ON MEMORY MANAGEMENT
022542  016100  140000                 MOV 140000(R1),R0               ;READ WORD
022546  005037  177572                 CLR SR0                         ;TURN OFF MEMORY MANAGEMENT
022552                                  POP R1
022552  012601                                  MOV (SP)+,R1
022554  000207                          RETURN
```

GLOBAL SUBROUTINES SECTION

```
                                        ;KTWRITE
                                        ;
                                        ;WRITE TO ONE WORD OF MEMORY USING KT HARDWARE. WORD IS SPECIFIED
                                        ;BY A BASE ADDRESS AND A BYTE OFFSET. THE BASE ADDRESS MUST BE
                                        ;PREVIOUSLY SPECIFIED BY A KTBASE CALL.
                                        ;
                                        ;INPUTS:
                                        ;       R0 - DATA TO BE WRITTEN
                                        ;       R1 - BYTE OFFSET INTO BUFFER
                                        ;OUTPUTS:
                                        ;       NONE

022556                                  KTWRITE:PUSH R1
022556  010146                                  MOV R1,-(SP)
022560  063701  002222                  ADD KTBASO,R1                   ;COMPUTE TOTAL OFFSET
022564                                  PUSH R1
022564  010146                                  MOV R1,-(SP)
022566  000301                          SWAB R1                         ;GET RID OF LOW BYTE
022570  042701  177400                  BIC #177400,R1
022574  063701  002220                  ADD KTBASA,R1                   ;ADD BASE ADDRESS
022600  006301                          ASL R1                          ;SHIFT TO POSITION FOR PAR REGISTER
022602  006301                          ASL R1
022604  010137  172354                  MOV R1,PAR6                     ;STORE IN PAR REGISTER
022610                                  POP R1
022610  012601                                  MOV (SP)+,R1
022612  042701  177400                  BIC #177400,R1
022616  012737  000001  177572          MOV #MM.EN,SR0                  ;TURN ON MEMORY MANAGEMENT
022624  010061  140000                  MOV R0,140000(R1)               ;WRITE WORD
022630  005037  177572                  CLR SR0                         ;TURN OFF MEMORY MANAGEMENT
022634                                  POP R1
022634  012601                                  MOV (SP)+,R1
022636  000207                          RETURN
```

```
      1                                 ;NXMI
      2                                 ;
      3                                 ;NON-EXISTANT MEMORY SERVICE ROUTINE
      4                                 ;
      5                                 ;INPUTS:
      6                                 ;       NXMAD SET TO ZERO
      7                                 ;OUTPUTS:
      8                                 ;       NXMAD SET TO ONES IF NON-EXISTANT TRAP OCCURED
      9
     10 022640                          BGNSRV NXMI
        022640                                                                  NXMI::
     11
     12 022640  012737  177777  002366          MOV #-1,NXMAD
     13
     14 022646                          ENDSRV
        022646                                                                  L10021:
        022646  000002                                                                 RTI
```

```
                                ;UDASRV
                                ;
                                ;UDA INTERRUPT SERVICE ROUTINE. MARKS UDA CONTROLLER TABLE THAT AN
                                ;INTERRUPT HAS BEEN RECEIVED.
                                ;
                                ;THIS ROUTINE IS CALLED BY A [JSR R0,UDASRV] INSTRUCTION FROM WITHIN
                                ;THE CONTROLLER TABLE. THE PC STORED IN R0 IS THE ADDRESS OF THE C.FLG
                                ;WORD IN THE CONTROLLER TABLE. THE STACK CONTAINS THE SAVED CONTENTS
                                ;OF R0 FOLLOWED BY THE INTERRUPTED PC AND PS.
                                ;
                                ;INPUTS:
                                ;       R0 - ADDRESS OF C.FLG WORD IN CONTROLLER TABLE
                                ;       STACK - SAVED CONTENTS OF R0
                                ;OUTPUTS:
                                ;       CT.CMD CLEARED AND CT.MSG SET IN C.FLG WORD OF CONTROLLER TABLE
                                ;       R0 - RESTORED FROM STACK
                                
022650                          BGNSRV UDASRV
022650                                                                  UDASRV::
                                ;       BIC #CT.CMD,(R0)                ;CLEAR CT.CMD
022650  052710  000010                  BIS #CT.MSG,(R0)                ;SET CT.MSG
022654                                  POP R0                          ;RESTORE R0
022654  012600                                  MOV (SP)+,R0
022656                          ENDSRV
022656                                                                  L10022:
022656  000002                                                                  RTI
```

```
                                ;SETTO
                                ;
                                ;SET TIMEOUT COUNTER TO SOME NUMBER OF SECONDS FROM CURRENT TIME.
                                ;
                                ;INPUTS:
                                ;       R0 - NUMBER OF SECONDS FOR TIMEOUT
                                ;       R1 - ADDRESS WHERE TWO WORD TIME TO BE PUT
                                ;OUTPUTS:
                                ;       R0 - CONTENTS DESTROYED
                                ;       R1 - INCREMENTED BY 2
                                ;
                                ;COMPUTE CLOCK TICKS TIL TIMEOUT
                                ;
022660                          SETTO:  PUSH <R2,R3>
022660  010246                                 MOV R2,-(SP)
022662  010346                                 MOV R3,-(SP)
022664  005002                          CLR R2                          ;CLEAR PRODUCT
022666  013703  002360                  MOV KW.HZ,R3                    ;GET MULTIPLICAND
022672  006200                  SETOO:  ASR R0                          ;SHIFT MULTIPLIER TO RIGHT
022674  103001                          BCC SETO1                       ;IF A ONE BIT SHIFTED OUT
022676  060302                          ADD R3,R2                       ;  ADD MULTIPLICAND TO PRODUCT
022700  006303                  SETO1:  ASL R3                          ;DOUBLE THE MULTIPLICAND
022702  005700                          TST R0
022704  001372                          BNE SETOO                       ;CONTINUE UNTIL MULTIPLIER IS ZERO
                                ;
                                ;GET CURRENT TIME
                                ;
022706  013700  002362          SETO2:  MOV KW.EL,R0                    ;GET TIME
022712  013703  002364                  MOV KW.EL+2,R3
022716  020037  002362                  CMP R0,KW.EL                    ;IF CHANGED DURING RETRIEVAL
022722  001371                          BNE SETO2                       ;  GET IT AGAIN
                                ;
                                ;ADD TIME TIL TIMEOUT
                                ;
022724  060200                          ADD R2,R0                       ;ADD
022726  005503                          ADC R3
                                ;
                                ;PUT RESULT IN STORAGE
                                ;
022730  010021                          MOV R0,(R1)+
022732  010311                          MOV R3,(R1)
                                ;
022734                                  POP <R3,R2>
022734  012603                                  MOV (SP)+,R3
022736  012602                                  MOV (SP)+,R2
022740  000207                          RETURN
```

```
                                ;UDAINT
                                ;
                                ;FUNCTIONAL DESCRIPTION:
                                ;       SUBROUTINE TO INITIALIZE A UDA AND BRING IT ON-LINE.
                                ;       ALL STEPS ARE CHECKED. AN ERROR MESSAGE IS REPORTED IF ANY ERROR
                                ;       DETECTED.
                                ;
                                ;INPUTS:
                                ;       R5 - ADDRESS OF CONTROLLER TABLE.
                                ;       R4 - LEN, INTI AND VECTOR FIELDS TO SEND TO UDA
                                ;IMPLICIT INPUTS:
                                ;       FFREE - FIRST FREE ADDRESS OF MEMORY. THIS ADDRESS IS GIVEN TO UDA
                                ;               AS START OF RING BUFFER.
                                ;       FSIZE - SIZE OF FREE MEMORY AVAILABLE IN WOPDS.
                                ;OUTPUTS:
                                ;       CONDITION Z - SET IF ANY ERROR REPORTED. CLEAR IF NO ERROR.
                                ;       R1 - SIZE OF RING BUFFER IN WORDS IF NO ERROR.
                                ;       R4 - ADDRESS OF UDAIP REGISTER IN UDA
                                ;       R5 - UNCHANGED.

                                ;CHECK IF ENOUGH FREE MEMORY FOR RING BUFFER

022742  010400                  UDAINT: MOV R4,R0                       ;GET MESSAGE LENGTH
022744  000300                          SWAB R0
022746  042700  177770                 BIC #177770,R0
022752  004737  023654                 CALL CLOG                       ;COMPUTE LOGARITHMIC VALUE
022756  010102                          MOV R1,R2                       ;SAVE RESULT IN R2
022760  010400                          MOV R4,R0                       ;GET COMMAND LENGTH
022762  000300                          SWAB R0
022764  006000                          ROR R0
022766  006000                          ROR R0
022770  006000                          ROR R0
022772  042700  177770                 BIC #177770,R0
022776  004737  023654                 CALL CLOG                       ;COMPUTE LOGARITHMIC VALUE
023002  060201                          ADD R2,R1                       ;ADD THE TWO RESULTS
023004  006301                          ASL R1                          ;MULTIPLY BY 2 WORDS PER RING
023006  062701  000002                 ADD #HC.ISZ/2,R1                ;ADD SPACE FOR INTERRUPT INDICATORS
023012  020137  002176                 CMP R1,FSIZE                    ;COMPARE WITH SIZE OF FREE MEMORY
023016  101402                          BLOS UDAI1
023020  000137  012710                 JMP FMERR                       ;FATAL ERROR IF NOT ENOUGH MEMORY
```

```
                                ;FILL HOST COMMUNICATION AREA WITH ALL ONES

023024  013702  002174  UDAI1:  MOV FFREE,R2             ;GET FIRST ADDRESS OF RING BUFFER
023030  010103                  MOV R1,R3               ;GET SIZE OF RING BUFFER
023032  012722  177777  UDAI1L: MOV #-1,(R2)+            ;WRITE ONES TO BUFFER
023036  005303                  DEC R3                  ;COUNT THE WORDS IN BUFFER
023040  003374                  BGT UDAI1L              ;LOOP UNTIL ENTIRE BUFFER WRITTEN

                                ;DO THE INITIALIZATION

023042  004737  023204          CALL UDAIST             ;DO FIRST THREE STEPS
023046  103454                  BCS UDAIEX              ;GET OUT IF UDA MICROCODE REPORTED FAILURE
023050  012364  000002          MOV (R3)+,2(R4)         ;WRITE NEXT WORD TO UDASA REGISTER
023054  012703  000310          MOV #200.,R3            ;GET TRY COUNTER
023060  016402  000002  UDAI1A: MOV 2(R4),R2           ;LOOK AT UDASA
023064  001410                  BEQ UDAI1C
023066  100005                  BPL UDAI1B
023070                          ERRDF 7,INTHD,INTR5
023070  104455                                                  TRAP    C$ERDF
023072  000007                                                  .WORD   7
023074  003263                                                  .WORD   INTHD
023076  012200                                                  .WORD   INTR5
023100  000437                  BR UDAIEX
023102  005303          UDAI1B: DEC R3
023104  001365                  BNE UDAI1A
023106  010264  000002  UDAI1C: MOV R2,2(R4)            ;WRITE 0 TO UDASA (PURGE)
023112  011402                  MOV (R4),R2             ;READ FROM UDAIP (POLL)
023114  004737  023514          CALL UDARSP             ;WAIT FOR STEP OR ERROR BIT
023120  103427                  BCS UDAIEX              ;GET OUT IF UDA MICROCODE REPORTED FAILURE

                                ;CHECK HOST COMMUNICATION AREA FOR ALL ZEROS

023122  013702  002174  UDAI2:  MOV FFREE,R2             ;GET FIRST ADDRESS OF RING BUFFER
023126  010103                  MOV R1,R3               ;GET SIZE OF RING BUFFER
023130  005722          UDAI2L: TST (R2)+               ;CHECK WORD IN BUFFER
023132  001003                  BNE UDAI2E              ;GO TO ERROR REPORTER IF NOT ZERO
023134  005303                  DEC R3                  ;COUNT THE WORDS IN BUFFER
023136  003374                  BGT UDAI2L              ;LOOP UNTIL ALL WORDS CHECKED
023140  000405                  BR UDAI3

023142                  UDAI2E: ERRDF 15,INTHD,INTBF    ;REPORT BUFFER NOT CLEARED
023142  104455                                                  TRAP    C$ERDF
023144  000017                                                  .WORD   15
023146  003263                                                  .WORD   INTHD
023150  012244                                                  .WORD   INTBF
023152  000412                  BR UDAIEX

                                ;SEND GO BIT TO UDASA REGISTER TO END INITIALIZATION

023154  016500  000006  UDAI3:  MOV C.BST(R5),R0          ;GET BURST VALUE
023160  006300                  ASL R0                  ;SHIFT TO POSITION
023162  006300                  ASL R0
023164  052700  000001          BIS #SA.GO,R0           ;SET THE GO BIT
023170  010064  000002          MOV R0,2(R4)            ;SEND TO UDA
023174  000244                  CLZ                     ;CLEAR Z AS NO ERROR INDICATION
023176  000207                  RETURN
```

```
                                ;ERROR RETURN
                                
023200  000264  UDAIEX: SEZ                     ;SET Z TO INDICATE ERROR OCCURRED
023202  000207          RETURN
```

```
                                ;UDAIST
                                ;
                                ;START THE INITIALIZATION PROCESS ON THE SELECTED UDA.
                                ;STOP BEFORE WRITING THE THIRD WORD SO UDA DOES NOT
                                ;ATTEMPT ANY UNIBUS TRANSFERS.
                                ;
                                ;INPUTS:
                                ;       R5 - ADDRESS OF CONTROLLER TABLE
                                ;       R4 - LEN, INTI AND VECTOR FIELDS TO SEND TO UDA

                                ;LOAD TABLE OF DATA TO SEND TO UDASA REGISTER

023204                          UDAIST: BREAK
023204  104422                                                                    TRAP    C$BRK
023206                                  PUSH R1
023206  010146                                  MOV R1,-(SP)
023210  052704  100000                  BIS #SA.STP,R4          ;SET STEP BIT IN DATA WORD
023214  010437  023406                  MOV R4,UDAID1           ;LOAD LENGTH AND INTERRUPT VECTOR
023220  013737  002174  023412          MOV FFREE,UDAID2        ;LOAD MEMORY ADDRESS
023226  062737  000004  023412          ADD #HC.MSG,UDAID2      ;  OF FIRST RESPONSE RING

                                ;START THE INITIALIZATION BY WRITING TO UDAIP REGISTER

023234  016504  000000                  MOV C.UADR(R5),R4       ;GET ADDRESS OF UDAIP REGISTER
023240  005037  002366                  CLR NXMAD               ;CLEAR MEMORY ERROR FLAG
023244                                  SETVEC #4,#NXMI,#PRI07  ;SET UP VECTOR 4
023244  012746  000340                                                          MOV     #PRI07,-(SP)
023250  012746  022640                                                          MOV     #NXMI,-(SP)
023254  012746  000004                                                          MOV     #4,-(SP)
023260  012746  000003                                                          MOV     #3,-(SP)
023264  104437                                                                  TRAP    C$SVEC
023266  062706  000010                                                          ADD     #10,SP
023272  005764  000002                          TST 2(R4)               ;ACCESS UDASA REGISTER
023276  005014                                  CLR (R4)                ;WRITE TO UDAIP
023300                                          CLRVEC #4               ;GIVE UP THE VECTOR
023300  012700  000004                                                          MOV     #4,R0
023304  104436                                                                  TRAP    C$CVEC
023306  005737  002366                          TST NXMAD               ;SEE IF A MEMORY ERROR OCCURRED
023312  001406                                  BEQ UDAISG
023314                                          ERRDF 21,0,INTR4
023314  104455                                                                  TRAP    C$ERDF
023316  000025                                                                  .WORD   21
023320  000000                                                                  .WORD   0
023322  012154                                                                  .WORD   INTR4
023324  000261                          SEC
023326  000424                          BR UDAISE

                                ;SET UP LOOP PARAMETERS TO EXECUTE THE FOUR STEPS OF INITIALIZATION

023330  012737  004000  023652  UDAISG: MOV #SA.S1,UDARSD       ;STORE RESPONSE MASK
023336  012703  023404                  MOV #UDAIDT,R3          ;AND INDEX TO TABLE

                                ;WAIT FOR AND CHECK RESPONSE DATA

023342  004737  023514          UDAISL: CALL UDARSP             ;WAIT FOR STEP OR ERROR BITS
023346  103414                          BCS UDAISE              ;EXIT IF ERROR
023350  004733                          CALL @(R3)+             ;CALL RESPONSE CHECKER FOR STEP
```

```
     44 023352  103412                          BCS UDAISE                      ;GET OUT IF ERROR
     45 023354  006337  023652                   ASL UDARSD                      ;SHIFT TO NEXT STEP BIT
     46 023360  032737  040000  023652           BIT #SA.S4,UDARSD               ;CHECK IF NOW AT STEP 4
     47 023366  001003                          BNE UDAISX                      ;GET OUT IF SO
     48 023370  012364  000002                   MOV (R3)+,2(R4)                 ;WRITE DATA TO UDASA REGISTER
     49 023374  000762                          BR UDAISL                       ;STAY IN LOOP
     50
     51 023376  000241                  UDAISX: CLC                             ;CLEAR CARRY FOR NO ERROR INDICATION
     52 023400                          UDAISE: POP R1
        023400  012601                                  MOV (SP)+,R1
     53 023402  000207                          RETURN
```

```
                                ;DATA TO BE SENT AND RECEIVED BY UDA INITIALIZATION
                                
023404  023422                  UDAIDT: .WORD UDAIR1                    ;FIRST WORD RESPONSE CHECK ROUTINE
023406  000000                  UDAID1: .WORD 0                         ;FIRST WORD TO SEND TO UDASA
023410  023430                          .WORD UDAIR2                    ;SECOND WORD RESPONSE CHECK ROUTINE
023412  000000                  UDAID2: .WORD 0                         ;SECOND WORD TO SEND TO UDASA
023414  023450                          .WORD UDAIR3                    ;THIRD WORD RESPONSE CHECK ROUTINE
023416  100000                  UDAID3: .WORD SA.TST                    ;THIRD WORD TO SEND TO UDASA
023420  023466                          .WORD UDAIR4                    ;FOURTH WORD RESPONSE CHECK ROUTINE
                                
                                ;RESPONSE CHECK FOR FIRST WORD FROM UDASA
                                ;CHECK FOR PROPER CONTROLLER TYPE
                                
023422  012701  004400          UDAIR1: MOV #SA.S1+SA.DI,R1             ;SET STEP ONE BIT
023426  000422                          BR UDAIRC                       ;NOW COMPARE
                                
                                ;RESPONSE CHECK FOR SECOND WORD FROM UDASA
                                ;CHECK FOR ECHO OF INTI AND VECTOR
                                
023430  013701  023406          UDAIR2: MOV UDAID1,R1                   ;GET WORD SENT TO UDASA
023434  000301                          SWAB R1                         ;GET HIGH 8 BITS
023436  042701  177400                  BIC #177400,R1
023442  052701  010000                  BIS #SA.S2,R1                   ;SET STEP 2 BIT
023446  000412                          BR UDAIRC                       ;NOW COMPARE
                                
                                ;RESPONSE CHECK FOR THIRD WORD FROM UDASA
                                ;CHECK FOR ECHO OF MESSAGE AND COMMAND RING LENGTHS
                                
023450  013701  023406          UDAIR3: MOV UDAID1,R1                   ;GET WORD SENT TO UDASA
023454  042701  177400                  BIC #177400,R1                  ;JUST LOW 8 BITS
023460  052701  020000                  BIS #SA.S3,R1                   ;SET STEP 3 BIT
023464  000403                          BR UDAIRC                       ;NOW COMPARE
                                
                                ;RESPONSE CHECK FOR FOURTH WORD FROM UDASA
                                ;CHECK FOR ECHO OF PURGE AND LFAIL BITS
                                
023466  010201                  UDAIR4: MOV R2,R1                       ;GET RESPONSE FROM UDA
023470  042701  137400                  BIC #^C<SA.S4+SA.MCV>,R1        ;KEEP MICROCODE VERSION AND STEP 4
                                
                                ;COMPARE EXPECTED DATA IN R1 WITH ACTUAL DATA IN R2
                                
023474  020102                  UDAIRC: CMP R1,R2                       ;COMPARE THE DATA
023476  001405                          BEQ UDAIRX                      ;EXIT IF COMPARED CORRECTLY
023500                                  ERRDF 6,INTHD,INTR3             ;REPORT ERROR
023500  104455                                                                  TRAP    C$ERDF
023502  000006                                                                  .WORD   6
023504  003263                                                                  .WORD   INTHD
023506  012066                                                                  .WORD   INTR3
023510  000261                          SEC
023512  000207                  UDAIRX: RETURN
```

```
                                ;UDARSP
                                ;
                                ;WAIT FOR UDA TO RESPOND WITH DATA IN UDASA REGISTER.
                                ;EITHER STEP BIT FROM MASK IN LOCATION UDARSD OR ERROR BIT
                                ;WILL CAUSE A TERMINATION.
                                ;AN ERROR MESSAGE WILL BE PRINTED IF THE UDA DOES NOT RESPOND
                                ;IN 10 SECONDS OR IF ERROR SETS.
                                ;
                                ;INPUTS:
                                ;       UDASRD - MASK OF STEP BIT TO LOOK FOR
                                ;       R5 - ADDRESS OF CONTROLLER TABLE
                                ;       R4 - ADDRESS OF UDAIP REGISTER
                                ;OUTPUTS:
                                ;       ERROR MESSAGE IF TIME OUT ON RESPONSE OR ERROR BIT SETS
                                ;       R2 - DATA FROM UDASA REGISTER
                                ;       CARRY SET IF ERROR BIT SETS OR TIME OUT
                                
023514                          UDARSP: PUSH R1
023514  010146                                MOV R1,-(SP)
023516  052737  100000  023652          BIS #SA.ERR,UDARSD              ;SET ERROR BIT IN MASK WORD
023524  012700  000012                  MOV #10.,R0                     ;SET UP FOR 10 SECOND TIMEOUT
023530  010501                          MOV R5,R1                       ;POINT TO COUNTER IN CONTROLLER TABLE
023532  062701  000040                  ADD #C.TO,R1
023536  004737  022660                  CALL SETTO
023542                                  POP R1
023542  012601                                MOV (SP)+,R1
023544  033764  023652  000002  UDARS1: BIT UDARSD,2(R4)                 ;LOOK AT ERROR AND STEP BIT
023552  001024                          BNE UDARS2                      ;BRANCH IF EITHER SET
023554                                  BREAK
023554  104422                                                                  TRAP    C$BRK
023556  005737  002352                  TST KW.CSR                      ;SEE IF CLOCK ON SYSTEM
023562  001770                          BEQ UDARS1
023564  023765  002364  000042          CMP KW.EL+2,C.TOH(R5)           ;CHECK IF TIME OUT OCCURRED
023572  101005                          BHI 1$
023574  001363                          BNE UDARS1
023576  023765  002362  000040          CMP KW.EL,C.TO(R5)
023604  103757                          BLO UDARS1
023606  016402  000002          1$:     MOV 2(R4),R2                    ;GET REGISTER CONTENTS
023612                                  ERRDF 4,INTHD,INTR1             ;REPORT TIME OUT ERROR
023612  104455                                                                  TRAP    C$ERDF
023614  000004                                                                  .WORD   4
023616  003263                                                                  .WORD   INTHD
023620  011756                                                                  .WORD   INTR1
023622  000407                          BR UDARSE
```

GLOBAL SUBROUTINES SECTION

```
                                ;CHECK IF ERROR BIT SET
                                
023624  016402  000002  UDARS2: MOV 2(R4),R2              ;GET REGISTER CONTENTS
023630  100006                 BPL UDARSX                 ;EXIT IF ERROR NOT SET
023632                         ERRDF 5,INTHD,INTR2        ;REPORT ERROR INFO
                                                                  TRAP    C$FRDF
023632  104455                                                    .WORD   5
023634  000005                                                    .WORD   INTHD
023636  003263                                                    .WORD   INTR2
023640  012022
023642  000261          UDARSE: SEC
023644  000207                 RETURN

                                ;NORMAL EXIT

023646  000241          UDARSX: CLC                        ;CLEAR CARRY AS NO ERROR INDICATION
023650  000207                 RETURN

                                ;LOCATION FOR STEP BIT MASK

023652  000000          UDARSD: .WORD 0                    ;LOAD BY CALLING ROUTINE
```

GLOBAL SUBROUTINES SECTION

```
                                ;CLOG
                                ;
                                ;COMPUTE LOGARITHMIC VALUE OF NUMBER TO BASE 2.
                                ;
                                ;INPUTS:
                                ;       R0 - LOGARITHM TO BE CONVERTED
                                ;OUTPUTS:
                                ;       R1 - VALUE OF 2 RAISED TO POWER OF INPUT NUMBER
                                ;
023654                          CLOG:   PUSH R0
023654  010046                                 MOV R0,-(SP)
023656  005001                          CLR R1                  ;SET UP ZERO START VALUE
023660  000261                          SEC                     ;WITH CARRY READY TO SHIFT IN
023662  006101                  CLOGLP: ROL R1                  ;SHIFT TO LEFT
023664  005300                          DEC R0                  ; UNTIL R0
023666  100375                          BPL CLOGLP              ; GOES NEGATIVE
023670                          ;       POP R0
023670  012600                                  MOV (SP)+,R0
023672  000207                          RETURN
```

```
                                        ;RDDLL
                                        ;
                                        ;READ DISK DRIVE DOWNLINE LOAD PROGRAM INTO MEMORY
                                        ;
                                        ;INPUTS:
                                        ;       DLLNAM - NAME OF PROGRAM IN RAD50 (TWO WORDS)
                                        ;OUTPUTS:
                                        ;       FREE MEMORY CONTAINING PROGRAM
                                        ;       CARRY CLEAR IF NO ERROR, CARRY SET IF PROGRAM NOT FOUND

023674  012701  000005        RDDLL:  MOV #5,R1                       ;TYPE OF PROGRAM IN DATA FILE
023700  004737  023734                CALL RDREC                      ;READ PROGRAM INTO MEMORY
023704  006101                        ROL R1                          ;PRESERVE CARRY STATE IN R1
023706  004737  023716                CALL CLOSEF                     ;  WHILE CLOSING THE DATA FILE
023712  006001                        ROR R1                          ;  AS NORMAL POSITION IS LOST
023714  000207                        RETURN
```

```
      1                                 ;CLOSEF
      2                                 ;
      3                                 ;CLOSE DATA FILE FOR DM PROGRAMS
      4                                 ;
      5                                 ;INPUTS:
      6                                 ;       FILOPN - ZERO IF FILE NOT OPEN
      7                                 ;OUTPUTS:
      8                                 ;       NONE
      9
     10 023716  005737  002256          CLOSEF: TST FILOPN                      ;SEE IF FILE CURRENTLY OPEN
     11 023722  001403                          BEQ 1$
     12 023724                                  CLOSE                           ; IF SO, CLOSE IT
        023724  104435                                                                  TRAP    C$CLOS
     13 023726  005037  002256                  CLR FILOPN                      ;AND MARK AS SO
     14 023732  000207                  1$:     RETURN
```

```
                                        ;RDREC
                                        ;
                                        ;READ A RECORD FROM THE INPUT FILE. PLACE DATA INTO FREE MEMORY.
                                        ;
                                        ;INPUTS:
                                        ;       R1 - FILE TYPE
                                        ;               1 - TEST 1 DM PROGRAM
                                        ;               2 - TEST 2 DM PROGRAM
                                        ;               3 - TEST 3 DM PROGRAM
                                        ;               4 - TEST 4 DM PROGRAM
                                        ;               5 - DRIVE DIAGNOSTIC DOWNLINE LOAD PROGRAM
                                        ;       DLLNAM - IF R1 CONTAINS 5, TWO WORDS AT THIS ADDRESS CONTAIN
                                        ;               NAME OF PROGRAM IN RAD50.
                                        ;OUTPUTS:
                                        ;       DATA FROM RECORD IN MEMORY
                                        ;       CARRY CLEAR IF NO ERROR, CARRY SET IF ERROR
                                        ;
023734                                  RDREC:  PUSH <R0,R1,R2,R3,R4,R5>
023734  010046                                          MOV R0,-(SP)
023736  010146                                          MOV R1,-(SP)
023740  010246                                          MOV R2,-(SP)
023742  010346                                          MOV R3,-(SP)
023744  010446                                          MOV R4,-(SP)
023746  010546                                          MOV R5,-(SP)
023750  005737  002256                                  TST FILOPN                      ;SEE IF FILE ALREADY OPEN
023754  001006                                          BNE RDST
023756                                                  OPEN #FNAME                     ;IF NOT, OPEN FILE NOW
023756  012700  002240                                                                          MOV     #FNAME,R0
023762  104434                                                                                  TRAP    C$OPEN
023764  005237  002256                                  INC FILOPN                      ;AND MARK AS OPEN
023770  005005                                          CLR R5                          ;CLEAR LOAD ADDRESS (SEARCH MODE)
023772                                  RDST:   BREAK                                   ;ALLOW PROGRAM TO BE INTERRUPTED
023772  104422                                                                                  TRAP    C$BRK
                                                                                        ;GETBYTE CALLS DON'T SEEM TO BREAK ON CONTROL-C!
023774                                          GETBYTE R4                              ;GET A BYTE
023774  104426                                                                                  TRAP    C$GETB
023776  110004                                                                                  MOVB    R0,R4
024000  005704                                  TST R4                          ;IF ZERO
024002  001773                                  BEQ RDST                        ; KEEP READING
024004  022704  000001                          CMP #1,R4                       ;WHEN NOT ZERO
024010  001153                                  BNE RWRDE1                      ; IT BETTER BE A ONE
024012                                          GETBYTE R0                      ; AND THE NEXT BYTE
024012  104426                                                                                  TRAP    C$GETB
024014  060004                                  ADD R0,R4
024016  005700                                  TST R0                          ; IF ZERO, PROCESS DATA
024020  001432                                  BEQ RDDAT
024022  020001                                  CMP R0,R1                       ;CHECK IF TYPE OF FILE LOOKING FOR
024024  103430                                  BLO RDDAT                       ;IF TOO SOON IN FILE, KEEP SEARCHING
024026  101132                                  BHI RDERR                       ;IF PAST TYPE, GIVE ERROR RETURN
024030  004737  024272                          CALL FWORD                      ;GET NEXT TWO WORDS
024034  013702  002254                          MOV FDATA,R2
024040  004737  024272                          CALL FWORD
024044                                          GETBYTE R0                      ;GET CHECKSUM
024044  104426                                                                                  TRAP    C$GETB
024046  060004                                  ADD R0,R4                       ;ADD TO COMPUTED SUM
024050  105704                                  TSTB R4                         ;SEE IF THIS SUM IS ZERO
024052  001137                                  BNE RWRDE3                      ;  IF NOT, REPORT CHECKSUM ERROR
```

```
                                        CMP R1,#5                       ;IF FILE TYPE IS A 5
                                        BNE 1$
                                        CMP DLLNAM,R2                   ;  MATCH THE PROGRAM NAME
                                        BNE RDST                        ;KEEP SEARCHING IF NOT DESIRED PROGRAM
                                        CMP DLLNAM+2,FDATA
                                        BNE RDST
                                1$:     MOV FFREE,R5                    ;GET STORAGE ADDRESS
                                        BR RDST                         ;SWITCH FROM SEARCH TO STORE MODE

                                RDDAT:  CALL FWORD                      ;READ BYTE COUNT
                                        MOV FDATA,R3                    ;SAVE IN R3
                                        CALL FWORD                      ;READ LOAD ADDRESS
                                        SUB #6,R3                       ;SUBTRACT BYTES ALREADY READ FROM BYTE COUNT
                                        BEQ RWORDT                      ;IF RESULT IS ZERO, THIS IS A
                                                                        ;TRANSFER BLOCK
                                        TST R5                          ;IF IN SEARCH MODE,
                                        BEQ 3$                          ;  BYPASS TRANSFER ADDRESS COMPUTATION
                                        MOV FDATA,R1                    ;GET LOAD ADDRESS
                                        ADD R3,R1                       ;ADD BYTES IN RECORD
                                        SUB #DMFRST,R1
                                        ADD R5,R1                       ;COMPUTE STORAGE ADDRESS
                                        BIT #1,R1                       ;CHECK IF ODD BYTE ADDRESS
                                        BEQ 1$                          ;IF SO,
                                        INC R1                          ;  INCREMENT
                                1$:     SUB FFREE,R1                    ;SEE IF MORE MEMORY NEEDS TO BE ALLOCATED
                                        BLOS 2$
                                        ROR R1                          ;REDUCE TO WORDS
                                        CALL ALOCM                      ;ALLOCATE THE MEMORY
                                2$:     MOV FDATA,R2                    ;GET LOAD ADDRESS
                                        SUB #DMFRST,R2
                                        ADD R5,R2
                                3$:     GETBYTE R0                      ;GET DATA BYTE
                                                                                TRAP    C$GETB
                                        TST R5                          ;IF IN SEARCH MODE,
                                        BEQ 4$                          ;  BYPASS DATA STORAGE
                                        MOVB R0,(R2)+                   ;STORE IN MEMORY
                                4$:     ADD R0,R4                       ;UPDATE CHECKSUM
                                        DEC R3                          ;COUNT THE BYTE
                                        BNE 3$                          ;GET THEM ALL
                                        GETBYTE R0                      ;GET CHECKSUM
                                                                                TRAP    C$GETB
                                        ADD R0,R4                       ;ADD
                                        TSTB R4                         ;IF CHECKSUM CORRECT,
                                        BEQ RDST                        ;  THEN GO READ NEXT RECORD
                                        BR RWRDE3                       ;  ELSE REPORT ERROR

                                RWORDT: GETBYTE R0                      ;READ CHECKSUM BYTE
                                                                                TRAP    C$GETB
                                        ADD R0,R4                       ,ADD TO COMPUTED CHECKSUM
                                        TSTB R4                         ;CHECK LOW BYTE OF SUM
                                        BNE RWRDE3                      ;BRANCH IF CHECKSUM ERROR
                                        TST R5                          ;IF IN SEARCH MODE,
                                        BEQ RDST                        ;  KEEP ON SEARCHING
                                        POP <R5,R4,R3,R2,R1,R0>
                                                MOV (SP)+,R5
                                                MOV (SP)+,R4
                                                MOV (SP)+,R3
```

GLOBAL SUBROUTINES SECTION

```
        024260  012602                          MOV (SP)+,R2
        024262  012601                          MOV (SP)+,R1
        024264  012600                          MOV (SP)+,R0
     96 024266  000241                  CLC
     97 024270  000207                  RETURN
     98
     99 024272                  FWORD:  GETBYTE R0                      ;READ A BYTE FROM FILE
        024272  104426                                                          TRAP    C$GETB
    100 024274  060004                  ADD R0,R4                       ;UPDATE CHECKSUM ERROR
    101 024276  110037  002254          MOVB R0,FDATA                   ;START TO BUILD WORD
    102 024302                          GETBYTE R0                      ;READ ANOTHER BYTE FROM FILE
        024302  104426                                                          TRAP    C$GETB
    103 024304  060004                  ADD R0,R4                       ;UPDATE CHECKSUM
    104 024306  110037  002255          MOVB R0,FDATA+1                 ;COMPLETE WORD
    105 024312  000207                  RETURN
    106
    107 024314  004737  023716  RDERR:  CALL CLOSEF                      ;CLOSE FILE AS POSITION IS LOST
    108 024320                          POP <R5,R4,R3,R2,R1,R0>
        024320  012605                          MOV (SP)+,R5
        024322  012604                          MOV (SP)+,R4
        024324  012603                          MOV (SP)+,R3
        024326  012602                          MOV (SP)+,R2
        024330  012601                          MOV (SP)+,R1
        024332  012600                          MOV (SP)+,R0
    109 024334  000261                  SEC                             ;ERROR RETURN, FILE NOT FOUND
    110 024336  000207                  RETURN
```

```
      1 024340                  RWRDE1: ERRSF  30001,RWRDEM,RWRDM1      ;START FRAME NOT FOUND
        024340  104454                                                          TRAP    C$ERSF
        024342  072461                                                          .WORD   30001
        024344  003310                                                          .WORD   RWRDEM
        024346  012370                                                          .WORD   RWRDM1
      2 024350                          DOCLN
        024350  104444                                                          TRAP    C$DCLN
      3 024352                  RWRDE3: ERRSF  30003,RWRDEM,RWRDM3      ;CHECKSUM ERROR
        024352  104454                                                          TRAP    C$ERSF
        024354  072463                                                          .WORD   30003
        024356  003310                                                          .WORD   RWRDEM
        024360  012436                                                          .WORD   RWRDM3
      4 024362                          DOCLN
        024362  104444                                                          TRAP    C$DCLN
```

```
                                        ;KW11I
                                        ;
                                        ;CLOCK INTERRUPT SERVICE ROUTINE
                                        ;
024364                                  BGNSRV KW11I
024364                                                                  KW11I::
024364  062737  000001  002362                ADD #1,KW.EL            ;COUNT THE INTERRUPT
024372  005537  002364                        ADC KW.EL+2
024376  012777  000103  155746                MOV #KWOUT.,@KW.CSR     ;RESTART THE CLOCK
024404                                  ENDSRV
024404                                                                  L10023:
024404  000002                                                                  RTI
```

GLOBAL SUBROUTINES SECTION

```
      1 024406                                BGNSRV  INTSRV          ; UDA INTERRUPT SERVER
        024406                                                                 INTSRV::
      2 024406  005237  002236                INC     INTRCV          ; FLAG INTERRUPT AS RECEIVED
      3 024412                                ENDSRV
        024412                                                       L10024:
        024412  000002                                                       RTI
```

GLOBAL SUBROUTINES SECTION

```
                                ;RESET
                                ;RESET ALL UNIBUS DEVICES THEN RESTART THE CLOCK IF IT IS PRESENT
                                ;
                                ;INPUTS:
                                ;       KW.CSR - ADDRESS OF CLOCK REGISTER. ZERO IF NO CLOCK PRESENT.
                                ;       KWOUT. = DATA TO SEND TO CLOCK REGISTER TO RESTART IT.
                                ;OUTPUTS:
                                ;       NONE
                                
024414                          RESET:  BREAK
024414  104422                                                                  TRAP    C$BRK
024416                                  BRESET                  ;RESET ALL DEVICES
024416  104433                                                                  TRAP    C$RESET
024420  005737  002352                  TST KW.CSR              ;SEE IF A CLOCK IS PRESENT
024424  001403                          BEQ RESETX
024426  012777  000103  155716          MOV #KWOUT.,@KW.CSR     ;START UP THE CLOCK
024434  000207                  RESETX: RETURN
```

```
                                ;RNTIME
                                ;
                                ;PRINT RUNTIME
                                ;
                                ;INPUTS:
                                ;       KW.EL - CONTAINS ELAPSED TIME
                                ;       KW.HZ - HERTZ OF CLOCK
                                ;OUTPUTS:
                                ;       THE FOLLOWING IS PRINTED IN BASIC OUTPUT:
                                ;         'RUNTIME HH:MM:SS'' AND LINE IS ENDED

024436                          RNTIME: PUSH <R0,R1,R2,R3,R4,R5>
024436  010046                                MOV R0,-(SP)
024440  010146                                MOV R1,-(SP)
024442  010246                                MOV R2,-(SP)
024444  010346                                MOV R3,-(SP)
024446  010446                                MOV R4,-(SP)
024450  010546                                MOV R5,-(SP)
024452  005737  002352                 TST KW.CSR                      ;CHECK IF A CLOCK PRESENT
024456  001432                          BEQ RNTIMX                      ;BRANCH IF NOT
024460  013703  002362                 MOV KW.EL,R3                    ;GET ELAPSED TIME
024464  013704  002364                 MOV KW.EL+2,R4
024470  013700  002360                 MOV KW.HZ,R0                    ;GET SPEED OF CLOCK
024474  004737  020070                 CALL DIVIDE                     ;COMPUTE SECONDS OF ELAPSED TIME
024500  012700  000074                 MOV #60.,R0                     ;NOW DIVIDE BY 60
024504  004737  020070                 CALL DIVIDE                     ; TO COMPUTE MINUTES
024510  010501                          MOV R5,R1                       ;SAVE REMAINDER IS SECONDS
024512  004737  020070                 CALL DIVIDE                     ;DIVIDE BY 60 AGAIN
024516                                  PRINTB #ERRMRT,R3,R5,R1         ;MINUTES IN R5
024516  010146                                  MOV     R1,-(SP)
024520  010546                                  MOV     R5,-(SP)
024522  010346                                  MOV     R3,-(SP)
024524  012746  006300                         MOV     #ERRMRT,-(SP)
024530  012746  000004                         MOV     #4,-(SP)
024534  010600                                  MOV     SP,R0
024536  104414                                  TRAP    C$PNTB
024540  062706  000012                         ADD     #12,SP
024544                          RNTIMX: PRINTB #ERRNL                   ;END THE LINE
024544  012746  005321                         MOV     #ERRNL,-(SP)
024550  012746  000001                         MOV     #1,-(SP)
024554  010600                                  MOV     SP,R0
024556  104414                                  TRAP    C$PNTB
024560  062706  000004                         ADD     #4,SP
024564                                  POP <R5,R4,R3,R2,R1,R0>         ;HOURS IN R3
024564  012605                                  MOV (SP)+,R5
024566  012604                                  MOV (SP)+,R4
024570  012603                                  MOV (SP)+,R3
024572  012602                                  MOV (SP)+,R2
024574  012601                                  MOV (SP)+,R1
024576  012600                                  MOV (SP)+,R0
024600  000207                          RETURN

024602                                  ENDMOD
                                .SBTTL
```

```
                                ;.TITLE MISCELLANEOUS SECTIONS   (4)
                                .SBTTL  REPORT CODING SECTION
 37
 38 024602                              BGNMOD
 39
 40                             ;++
 41                             ; THE REPORT CODING SECTION CONTAINS THE
 42                             ; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
 43                             ;--
 44
 45 024602                              BGNRPT
    024602                                                                  L$RPT::
 46
 58 024602                              PUSH <R0,R1,R2,R3,R4,R5>
    024602  010046                             MOV R0,-(SP)
    024604  010146                             MOV R1,-(SP)
    024606  010246                             MOV R2,-(SP)
    024610  010346                             MOV R3,-(SP)
    024612  010446                             MOV R4,-(SP)
    024614  010546                             MOV R5,-(SP)
 59 024616                              PRINTS #RPTMSG,TNUM             ;PRINT TEST NUMBER
    024616  013746  002226                                                      MOV     TNUM,-(SP)
    024622  012746  025410                                                      MOV     #RPTMSG,-(SP)
    024626  012746  000002                                                      MOV     #2,-(SP)
    024632  010600                                                              MOV     SP,R0
    024634  104416                                                              TRAP    C$PNTS
    024636  062706  000006                                                      ADD     #6,SP
 60 024642  005737  002352              TST KW.CSR                      ;CHECK IF A CLOCK IS PRESENT
 61 024646  001432                     BEQ RPTEX                       ;BRANCH IF NOT
 62 024650  013703  002362              MOV KW.EL,R3                    ;GET ELAPSED TIME
 63 024654  013704  002364              MOV KW.EL+2,R4
 64 024660  013700  002360              MOV KW.HZ,R0                    ;GET SPEED OF CLOCK
 65 024664  004737  020070              CALL DIVIDE                     ;COMPUTE SECONDS OF ELAPSED TIME
 66 024670  012700  000074              MOV #60.,R0                     ;NOW DIVIDE BY 60
 67 024674  004737  020070              CALL DIVIDE                     ; TO COMPUTE MINUTES
 68 024700  010501                     MOV R5,R1                       ;SAVE REMAINDER AS SECONDS
 69 024702  004737  020070              CALL DIVIDE                     ;DIVIDE BY 60 AGAIN
 70 024706                              PRINTS #RPTMS2,R3,R5,R1         ;MINUTES IN R5, HOURS IN R3
    024706  010146                                                              MOV     R1,-(SP)
    024710  010546                                                              MOV     R5,-(SP)
    024712  010346                                                              MOV     R3,-(SP)
    024714  012746  025444                                                      MOV     #RPTMS2,-(SP)
    024720  012746  000004                                                      MOV     #4,-(SP)
    024724  010600                                                              MOV     SP,R0
    024726  104416                                                              TRAP    C$PNTS
    024730  062706  000012                                                      ADD     #12,SP
 71 024734                      RPTEX:  PRINTS #ERRNL                   ;END THE LINE
    024734  012746  005321                                                      MOV     #ERRNL,-(SP)
    024740  012746  000001                                                      MOV     #1,-(SP)
    024744  010600                                                              MOV     SP,R0
    024746  104416                                                              TRAP    C$PNTS
    024750  062706  000004                                                      ADD     #4,SP
 72 024754  023727  002226  000004            CMP TNUM,#4                     ;IF NOT TEST 4
 73 024762  001160                          BNE RPTXX                       ;  THAT IS ALL
 74 024764                                  PRINTS #RPTMSH
    024764  012746  025500                                                      MOV     #RPTMSH,-(SP)
    024770  012746  000001                                                      MOV     #1,-(SP)
```

```
        024774  010600                                                          MOV     SP,R0
        024776  104416                                                          TRAP    C$PNTS
        025000  062706  000004                                                  ADD     #4,SP
     75 025004                          PRINTS #RPTMH2
        025004  012746  025615                                                  MOV     #RPTMH2,-(SP)
        025010  012746  000001                                                  MOV     #1,-(SP)
        025014  010600                                                          MOV     SP,R0
        025016  104416                                                          TRAP    C$PNTS
        025020  062706  000004                                                  ADD     #4,SP
     76 025024  013705  002204          MOV CTABS,R5                    ;GET ADDRESS OF FIRST CONTROLLER TABLE
     77 025030  005765  000002  RPTCT:  TST C.UNIT(R5)                  ;SEE IF CONTROLLER AVAILABLE FOR TESTING
     78 025034  100527                  BMI RPTCTN
     79 025036                                  ASSUME CT.AVL EQ BIT15
     80 025036  010504                  MOV R5,R4                       ;COMPUTE ADDRESS OF DRIVE TABLE POINTERS
     81 025040  062704  000020          ADD #C.DR0,R4
     82 025044  012703  000001          MOV #1.,R3                      ;GET COUNT OF DRIVES
     83 025050  012401          RPTDT:  MOV (R4)+,R1                    ;LOOK AT POINTER
     84 025052  001520                  BEQ RPTCTN                      ;GO TO NEXT IF NO TABLE
     85 025054  005761  000002          TST D.UNIT(R1)                  ;SEE IF DRIVE AVAILABLE
     86 025060  100511                  BMI RPTDTN
     87 025062                                  ASSUME DT.AVL EQ BIT15
     88 025062                          PRINTS #RPTMSD,D.UNIT(R1),(R1)
        025062  011146                                                          MOV     (R1),-(SP)
        025064  016146  000002                                                  MOV     D.UNIT(R1),-(SP)
        025070  012746  025722                                                  MOV     #RPTMSD,-(SP)
        025074  012746  000003                                                  MOV     #3,-(SP)
        025100  010600                                                          MOV     SP,R0
        025102  104416                                                          TRAP    C$PNTS
        025104  062706  000010                                                  ADD     #10,SP
     89 025110                          PUSH <R1,R3,R4,R5>
        025110  010146                          MOV R1,-(SP)
        025112  010346                          MOV R3,-(SP)
        025114  010446                          MOV R4,-(SP)
        025116  010546                          MOV R5,-(SP)
     90 025120  016102  000200          MOV D.SERN(R1),R2               ;GET SERIAL NUMBER
     91 025124  016103  000202          MOV D.SERN+2(R1),R3
     92 025130  016104  000204          MOV D.SERN+4(R1),R4
     93 025134  012700  002260          MOV #TEMP,R0                    ;PLACE 18 SPACE CHARACTERS INTO
     94 025140  012701  000022          MOV #18.,R1                     ;  TEMP STORAGE
     95 025144  112720  000040    1$:   MOVB #' ,(R0)+
     96 025150  005301                  DEC R1
     97 025152  001374                  BNE 1$
     98 025154  005010                  CLR (R0)                        ;THEN A NULL CHARACTER
     99 025156  004737  025344    2$:   CALL DIV10                      ;DIVIDE BY 10
    100 025162  062705  000060          ADD #'0,R5                      ;CONVERT TO ASCII CHARACTER
    101 025166  110540                  MOVB R5,-(R0)                   ;PUT DIGIT INTO TEMP STORAGE
    102 025170  010201                  MOV R2,R1                       ;SEE IF QUOTIENT IS ZERO
    103 025172  050301                  BIS R3,R1
    104 025174  050401                  BIS R4,R1
    105 025176  001367                  BNE 2$                          ;IF NOT, DIVIDE AGAIN
    106 025200                          POP <R5,R4,R3,R1>
        025200  012605                          MOV (SP)+,R5
        025202  012604                          MOV (SP)+,R4
        025204  012603                          MOV (SP)+,R3
        025206  012601                          MOV (SP)+,R1
    107 025210                          PRINTS #RPTMD1,#TEMP,D.SEEK(R1),D.XFRR(R1),D.XFRW(R1)
        025210  016146  000164                                                  MOV     D.XFRW(R1),-(SP)
```

```
        025214  016146  000166                                                          MOV     D.XFRR(R1),-(SP)
        025220  016146  000174                                                          MOV     D.SEEK(R1),-(SP)
        025224  012746  002260                                                          MOV     #TEMP,-(SP)
        025230  012746  025742                                                          MOV     #RPTMD1,-(SP)
        025234  012746  000005                                                          MOV     #5,-(SP)
        025240  010600                                                                  MOV     SP,R0
        025242  104416                                                                  TRAP    C$PNTS
        025244  062706  000014                                                          ADD     #14,SP
    108 025250                          PRINTS #RPTMD2,D.HERR(R1),D.SERR(R1),D.ECCC(R1)
        025250  016146  000176                                                          MOV     D.ECCC(R1),-(SP)
        025254  016146  000172                                                          MOV     D.SERR(R1),-(SP)
        025260  016146  000170                                                          MOV     D.HERR(R1),-(SP)
        025264  012746  025772                                                          MOV     #RPTMD2,-(SP)
        025270  012746  000004                                                          MOV     #4,-(SP)
        025274  010600                                                                  MOV     SP,R0
        025276  104416                                                                  TRAP    C$PNTS
        025300  062706  000012                                                          ADD     #12,SP
    109 025304                                  ASSUME D.DRV EQ 0
    110 025304                                  ASSUME C.UADR EQ 0
    111 025304  005203                  RPTDTN: INC R3                          ;COUNT THE DRIVE TABLES
    112 025306  020327  000010                  CMP R3,#8.
    113 025312  002656                          BLT RPTDT                       ;REPEAT FOR ALL DRIVE TABLES
    114 025314  062705  000046          RPTCTN: ADD #C.SIZE,R5                  ;GO TO NEXT CONTROLLER TABLE
    115 025320  005715                          TST (R5)
    116 025322  001242                          BNE RPTCT
    117 025324                          RPTXX:  POP <R5,R4,R3,R2,R1,R0>
        025324  012605                                  MOV (SP)+,R5
        025326  012604                                  MOV (SP)+,R4
        025330  012603                                  MOV (SP)+,R3
        025332  012602                                  MOV (SP)+,R2
        025334  012601                                  MOV (SP)+,R1
        025336  012600                                  MOV (SP)+,R0
    118 025340                              EXIT    RPT
        025340  000167                                                                  .WORD   J$JMP
        025342  000450                                                                  .WORD   L10025-2-.
    119
    120 025344                          DIV10:  PUSH R1                         ;DIVIDEND IS IN <R2,R3,R4>
        025344  010146                                  MOV R1,-(SP)
    121 025346  012701  000060                  MOV #48.,R1                     ;SET UP SHIFT COUNT
    122 025352  005005                          CLR R5                          ;START WITH ZERO REMAINDER
    123 025354  006302                  1$:     ASL R2                          ;SHIFT LEFT INTO R5
    124 025356  006103                          ROL R3
    125 025360  006104                          ROL R4
    126 025362  006105                          ROL R5
    127 025364  022705  000012                  CMP #10.,R5                     ;SILL DIVISOR GO INTO REMAINDER?
    128 025370  101003                          BHI 2$                          ;ONLY SUBTRACT IF IT WILL
    129 025372  162705  000012                  SUB #10.,R5                     ;SUBTRACT DIVISOR
    130 025376  005202                          INC R2                          ;PUT A ONE INTO QUOTIENT
    131 025400  005301                  2$:     DEC R1                          ;COUNT THE SHIFTS
    132 025402  001364                          BNE 1$
    133 025404                                  POP R1                          ;RETURN WITH QUOTIENT IN
        025404  012601                                  MOV (SP)+,R1
    134 025406  000207                          RETURN                          ;  <R2,R3,R4> AND REMAINDER IN R5
    135
    147 025410      045     116     045 RPTMSG: .ASCIZ\%N%ATEST %D1%A IN PROGRESS.\
    148 025444      045     101     040 RPTMS2: .ASCIZ\%A  RUNTIME %D3%A:%Z2%A:%Z2\
    149 025500      045     116     045 RPTMSH: .ASCIZ\%N%AUNIT DRIVE      SERIAL-NUMBER SEEKS MBYTES  MBYTES  HARD   SOFT    ECC%N
```

REPORT CODING SECTION

```
    150 025615      045     101     040  RPTMH2: .ASCIZ  \%A                                    X1000  READ  WRITTEN ERRORS ERRORS%N\
    151 025722      045     123     062  RPTMSD. .ASCIZ\%S2%D2%S3%D3%S1\
    152 025742      045     124     045  RPTMD1: .ASCIZ\%T%S1%D5%S2%D5%S3%D5%S2\
    153 025772      045     104     065  RPTMD2: .ASCIZ\%D5%S2%D5%S1%D5%N\
    154                                          .EVEN
    155
    156 026014                               ENDRPT
        026014                                                          L10025:
        026014  104425                                                        TRAP    C$RPT
```

PROTECTION TABLE

```
                                .SBTTL  PROTECTION TABLE

                                ;++
                                ; THIS TABLE IS USED BY THE RUNTIME SERVICES
                                ; TO PROTECT THE LOAD MEDIA.
                                ;--

026016                                  BGNPROT
026016                                                                  L$PROT::

026016  177777                          -1                      ;OFFSET INTO P-TABLE FOR CSR ADDRESS
026020  177777                          -1                      ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
026022  177777                          -1                      ;OFFSET INTO P-TABLE FOR DRIVE NUMBER

026024                                  ENDPROT
```

```
                                .SBTTL  INITIALIZE SECTION

                                ;++
                                ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
                                ; AT THE BEGINNING OF EACH PASS.
                                ;--

026024                                  BGNINIT
026024                                                                          L$INIT::

                                                .REPT 0
                                    IF HERE FROM CONTINUE COMMAND
                                        THEN
                                            SET ICONT BIT IN IFLAGS
                                    ENDIF
                                    IF HERE FROM RESTART COMMAND
                                        THEN
                                            SET IREST BIT IN IFLAGS
                                    ENDIF
                                    IF HERE FROM POWER FAIL RESTART
                                        THEN
                                            RESET ALL UNITS
                                            INITIALIZE MEMORY MANAGEMENT REGISTERS
                                            PRINT STATISTICAL REPORT
                                            DELAY ONE MINUTE
                                    ENDIF
                                    IF HERE FROM START COMMAND
                                        THEN
                                            RESET ALL UNITS
                                            INITIALIZE MEMORY MANAGEMENT REGISTERS
                                            ESTABLISH FREE MEMORY
                                            CLEAR TNUM
                                            SET ISTRT BIT IN IFLAGS, CLEAR OTHER BITS
                                            INITIALIZE CLOCK
                                            BUILD TABLES
                                        ELSE
                                            CHECK TABLES FOR ADDED OR DROPPED UNITS
                                .ENDF
                                                .ENDR

026024                                  READEF #EF.CONTINUE             ;HERE FROM CONTINUE COMMAND?
026024  012700  000036                                                          MOV     #EF.CONTINUE,R0
026030  104447                                                                  TRAP    C$REFG
026032                                  BNCOMPLETE INIT1                ;JUMP IF NOT
026032  103003                                                                  BCC     INIT1
026034  052737  000002  002224          BIS #ICONT,IFLAGS               ;SET CONTINUE BIT IN FLAG RECORD
026042                          INIT1:  READEF #EF.RESTART              ;LOOK AT EVENT FLAGS
026042  012700  000037                                                          MOV     #EF.RESTART,R0
026046  104447                                                                  TRAP    C$REFG
026050                                  BNCOMPLETE INIT1A               ;SET IREST BIT IN IFLAGS
026050  103003                                                                  BCC     INIT1A
026052  052737  000004  002224          BIS #IREST,IFLAGS               ;  IF HERE FROM RESTART COMMAND
026060                          INIT1A: READEF #EF.PWR                  ;HERE FROM POWER RESTART?
026060  012700  000034                                                          MOV     #EF.PWR,R0
026064  104447                                                                  TRAP    C$REFG
026066                                  BNCOMPLETE INIT2                ;JUMP IF SET
026066  103052                                                                  BCC     INIT2
```

INITIALIZE SECTION

```
     71 026070  004737  024414                    CALL RESET                      ;RESET ALL UNITS
     72 026074  004737  022240                    CALL INITKT                     ;INITIALIZE MEMORY MANAGEMENT REGISTERS
     73 026100                                    DORPT                           ;PRINT A STATISTICAL REPORT
        026100  104424                                                                  TRAP    C$DRPT
     74 026102  005737  002352                    TST KW.CSR                      ;CHECK IF CLOCK ON SYSTEM
     75 026106  001421                            BEQ POWDLY                      ;IF NOT, USE A LOOP DELAY
     76 026110  012700  000074                    MOV #60.,R0                     ;SET UP A TIME OUT OF 60 SECONDS
     77 026114  012701  002372                    MOV #STIME,R1
     78 026120  004737  022660                    CALL SETTO
     79 026124                          POWDLC:   BREAK
        026124  104422                                                                  TRAP    C$BRK
     80 026126  023737  002374  002364            CMP STIME+2,KW.EL+2             ;CHECK IF TIME ELAPSED
     81 026134  101027                            BHI INIT2
     82 026136  001372                            BNE POWDLC
     83 026140  023737  002372  002362            CMP STIME,KW.EL
     84 026146  103766                            BLO POWDLC
     85 026150  000421                            BR INIT2
     86 026152  012701  002400          POWDLY:   MOV #2400,R1                    ;DELAY ONE MINUTE FOR DISK
     87 026156                          POWDLL:   DELAY 250.                      ; UNITS TO POWER UP
        026156  012727  000372                                                          MOV     #250.,(PC)+
        026162  000000                                                                  .WORD   0
        026164  013727  002116                                                          MOV     L$DLY,(PC)+
        026170  000000                                                                  .WORD   0
        026172  005367  177772                                                          DEC     -6(PC)
        026176  001375                                                                  BNE     .-4
        026200  005367  177756                                                          DEC     -22(PC)
        026204  001367                                                                  BNE     .-20
     88 026206                                    BREAK                           ; WHILE RESPONDING TO ^C
        026206  104422                                                                  TRAP    C$BRK
     89 026210  005301                            DEC R1                          ; EVERY0.025 SECOND
     90 026212  003357                            BGT POWDLY
     91 026214                          INIT2:    READEF #EF.START                ;HERE FROM START COMMAND?
        026214  012700  000040                                                          MOV     #EF.START,R0
        026220  104447                                                                  TRAP    C$REFG
     92 026222                                    BCOMPLETE INIT3                 ;JUMP IF NOT
        026222  103467                                                                  BCS     INIT3
     93
     94                                 ;SET NOT AVAILABLE BITS IN ALL CONTROLLER AND DRIVE TABLES
     95
     96 026224  013705  002204                    MOV CTABS,R5                    ;GET FIRST CONTROLLER TABLE ADDRESS
     97
     98 026230  052765  100000  000002  INITC1:   BIS #CT.AVL,C.UNIT(R5)          ;SET BIT IN CONTROLLER TABLE
     99 026236  010502                            MOV R5,R2                       ;GET POINTER TO DRIVE TABLES
    100 026240  062702  000020                    ADD #C.DRO,R2
    101 026244  012703  000010                    MOV #8.,R3                      ;GET COUNT OF DRIVE TABLES
    102 026250  012200                  INITC2:   MOV (R2)+,R0                    ;CHECK IF ANY MORE DRIVE TABLES
    103 026252  001405                            BEQ INITC3
    104 026254  052760  100000  000002            BIS #DT.AVL,D.UNIT(R0)          ;SET BIT IN DRIVE TABLE
    105 026262  005303                            DEC R3
    106 026264  003371                            BGT INITC2
    107 026266  062705  000046          INITC3:   ADD #C.SIZE,R5                  ;MOVE TO NEXT CONTROLLER TABLE
    108 026272  005715                            TST (R5)                        ;IS THERE A NEXT ONE?
    109 026274  001355                            BNE INITC1                      ;IF SO, CLEAR THE BITS THERE
    110
    111                                 ;NOW GET EACH P-TABLE AND CLEAR NOT AVAILABLE BITS
    112
    113 026276  005003                            CLR R3                          ;START WITH UNIT 0
```

```
    114 026300                          INITC4: GPHARD R3,R0                    ;GET HW P-TABLE
        026300  010300                                                                  MOV     R3,R0
        026302  104442                                                                  TRAP    C$GPHRD
    115 026304                                  BNCOMPLETE INITC7                   ;GO AROUND IF NOT AVAILABLE
        026304  103030                                                                  BCC     INITC7
    116 026306  013705  002204              INITC5: MOV CTABS,R5                    ;GET FIRST CONTROLLER TABLE
    117 026312  021015                          CMP (R0),(R5)                       ;COMPARE UDA ADDRESSES
    118 026314  001411                          BEQ INITCC
    119 026316  062705  000046                  ADD #C.SIZE,R5                      ;LOOK AT NEXT CONTROLLER TABLE
    120 026322  005715                          TST (R5)                            ;IF THERE IS ANY
    121 026324  001372                          BNE INITC5
    122 026326                                  ERRSF 1,BADT
        026326  104454                                                                  TRAP    C$ERSF
        026330  000001                                                                  .WORD   1
        026332  003513                                                                  .WORD   BADT
        026334  000000                                                                  .WORD   0
    123 026336                          INITE1: DOCLN
        026336  104444                                                                  TRAP    C$DCLN
    124 026340  016001  000010          INITCC: MOV HD.LDR(R0),R1
    125 026344  004737  017414                  CALL GTDRVT
    126 026350  001372                          BNE INITE1
    127 026352  042765  100000  000002  INITC6: BIC #CT.AVL,C.UNIT(R5)              ;CLEAR BIT IN CONTROLLER TABLE
    128 026360  042764  100000  000002          BIC #DT.AVL,D.UNIT(R4)              ;CLEAR BIT IN DRIVE TABLE
    129 026366  005203                  INITC7: INC R3                              ;INCREMENT UNIT NUMBER
    130 026370  023703  002012                  CMP L$UNIT,R3                       ;CHECK IF GOT ALL TABLES
    131 026374  003341                          BGT INITC4                          ;IF NOT, GO GET ANOTHER
    132 026376  000137  027412                  JMP INITXX                          ;EXIT THE INIT CODE
```

INITIALIZE SECTION

```
                                INIT3:  BRESET                          ;RESET ALL UNITS
026402  104433                                                                  TRAP    C$RESET
                                        CALL INITKT                     ;INITIALIZE MEMORY MANAGEMENT
                                        MEMORY FFREE                    ;RESET START OF FREE MEMORY
026410  104431                                                                  TRAP    C$MEM
026412  010037  002174                                                          MOV     R0,FFREE
026416  017737  153552  002176          MOV @FFREE,FSIZE                ;RESET SIZE OF FREE MEMORY
026424  005037  002226                  CLR TNUM                        ;INITIALIZE TEST NUMBER TO NO TEST RUNNING
026430  012737  000010  002224          MOV #ISTRT,IFLAGS               ;SET START FLAG FOR TEST 4

                                ;INITIALIZE CLOCK

        000103                           KWOUT.=103                     ;DATA TO SEND TO KW11 TO START CLOCK
026436  005037  002362                  CLR KW.EL                       ;CLEAR ELAPSED TIME
026442  005037  002364                  CLR KW.EL+2
026446                                  CLOCK L,R0                      ;SEE IF AN L CLOCK PRESENT
026446  012700  000114                                                          MOV     #'L,R0
026452  104462                                                                  TRAP    C$CLCK
026454                                  BCOMPLETE KYES
026454  103417                                                                  BCS     KYES
026456                                  CLOCK P,R0                      ;SEE IF A P CLOCK PRESENT
026456  012700  000120                                                          MOV     #'P,R0
026462  104462                                                                  TRAP    C$CLCK
026464                                  BCOMPLETE KYES
026464  103413                                                                  BCS     KYES
026466  005037  002352                  CLR KW.CSR                      ;IF NEITHER, CLEAR CSR STORAGE WORD
026472                                  PRINTF #NOCLOCK
026472  012746  011044                                                          MOV     #NOCLOCK,-(SP)
026476  012746  000001                                                          MOV     #1,-(SP)
026502  010600                                                                  MOV     SP,R0
026504  104417                                                                  TRAP    C$PNTF
026506  062706  000004                                                          ADD     #4,SP
026512  000444                          BR KNO
026514  012037  002352          KYES:   MOV (R0)+,KW.CSR                ;STORE DATA RETURNED
026520  012037  002354                  MOV (R0)+,KW.BRL
026524  012037  002356                  MOV (R0)+,KW.VEC
026530  012037  002360                  MOV (R0)+,KW.HZ
026534                                  SETVEC KW.VEC,#KW11I,KW.BRL     ;SET THE VECTOR
026534  013746  002354                                                          MOV     KW.BRL,-(SP)
026540  012746  024364                                                          MOV     #KW11I,-(SP)
026544  013746  002356                                                          MOV     KW.VEC,-(SP)
026550  012746  000003                                                          MOV     #3,-(SP)
026554  104437                                                                  TRAP    C$SVEC
026556  062706  000010                                                          ADD     #10,SP
026562  012777  000103  153562          MOV #KWOUT.,@KW.CSR             ;START THE CLOCK
026570  013702  002360                  MOV KW.HZ,R2                    ;INITIALIZE TIME FOR FIRST
026574  000302                          SWAB R2                         ; STATISTICAL REPORT
026576  006302                          ASL R2                          ;  FOR ABOUT 15 MINUTES FROM NOW
026600  006302                          ASL R2
026602  063702  002362                  ADD KW.EL,R2
026606  013737  002364  002374          MOV KW.EL+2,STIME+2
026614  010237  002372                  MOV R2,STIME
026620  005537  002374                  ADC STIME+2
026624                          KNO:
```

```
                                        ;INITIALIZE CONTROLLER TABLE STORAGE WITH A WORD OF ZEROS
                                        ;
026624  013737  002174  002204                MOV FFREE,CTABS                 ;STORE START OF CONTROLLER TABLES
026632  005077  153346                        CLR @CTABS                      ;ZEROS MARKS END CONTROLLER TABLES
026636  005037  002206                        CLR CTRLRS                      ;CLEAR CONTROLLER COUNT
                                        ;
                                        ;GET A P-TABLE FROM DRS
                                        ;
026642  005002                          INIT4: CLR R2                          ;LOGICAL UNIT NUMBER IN R2
026644                                         GPHARD R2,R0                    ;GET POINTER TO A P-TABLE
026644  010200                                                                         MOV     R2,R0
026646  104442                                                                         TRAP    C$GPHRD
026650                                         BNCOMPLETE NXTTAB               ;IGNORE IF NO TABLE RETURNED
026650  103064                                                                         BCC     NXTTAB
                                        ;
                                        ;SEE IF A CONTROLLER TABLE ALREADY EXISTS FOR CONTROLLER IN P-TABLE
                                        ;
026652  013703  002204                  INIT5: MOV CTABS,R3                   ;GET ADDRESS OF CONTROLLER TABLES
026656  005713                                 TST (R3)                       ;CHECK IF ANY MORE TABLES
026660  001405                                 BEQ NEWTAB                     ;BUILD NEW TABLE IF FOUND ZERO WORD
026662  021013                                 CMP (R0),(R3)                  ;CHECK IF SAME UNIBUS ADDRESS
026664                                                 ASSUME C.UADR EQ 0
026664                                                 ASSUME HO.UBA EQ 0
026664  001437                                 BEQ SAMTAB                     ;CHECK TABLE IF ALREADY EXISTS
026666  062703  000046                         ADD #C.SIZE,R3                 ;MOVE TO NEXT TABLE
026672  000771                                 BR INIT5
```

INITIALIZE SECTION

```
                                        ;BUILD A CONTROLLER TABLE
                                        
026674  012701  000023    NEWTAB: MOV #C.SIZE/2,R1        ;GET WORDS IN CONTROLLER TABLE
026700  004737  012722            CALL ALOCM              ;ALLOCATE SPACE FOR IT
026704  011021                    MOV (R0),(R1)+          ;STORE UNIBUS ADDRESS
026706  010221                    MOV R2,(R1)+            ;UNIT NUMBER
026710  016004  000004            MOV H0.BRL(R0),R4       ;GET BR LEVEL
026714  000304                    SWAB R4                 ;SWAP TO HIGH BYTE
026716  006104                    ROL R4                  ;SHIFT ONE MORE TO LEFT
026720  056004  000002            BIS H0.VEC(R0),R4       ;ADD VECTOR ADDRESS
026724  010421                    MOV R4,(R1)+            ;  TO TABLE
026726  016021  000006            MOV H0.BST(R0),(R1)+
026732  012721  004037            MOV #4037,(R1)+         ;PUT [JSR R0,UDASRV]
026736  012721  022650            MOV #UDASRV,(R1)+       ;  INTO TABLE
026742  012703  000015            MOV #13.,R3             ;CLEAR POINTERS TO DRIVE TABLES,
026746  005021            INIT7:  CLR (R1)+               ; TIMEOUT COUNTER, FLAGS, REF. NUMBER
026750  005303                    DEC R3
026752  001375                    BNE INIT7               ;LOOP TIL ALL CLEARED
026754  005237  002206            INC CTRLRS              ;COUNT THE CONTROLLER
026760  005011                    CLR (R1)                ;CLEAR TABLE END MARKER
026762  000417                    BR NXTTAB               ;NOW GO TO NEXT P-TABLE
```

INITIALIZE SECTION

```
                                        ;SHOULD BE SAME CONTROLLER, CHECK THAT OTHER PARAMETERS MATCH
                                        
026764  016004  000004          SAMTAB: MOV HO.BRL(R0),R4               ;GET BR LEVEL FROM P-TABLE
026770  000304                          SWAB R4                         ;SWAP TO HIGH BYTE
026772  006104                          ROL R4                          ;SHIFT ONE MORE TO LEFT
026774  056004  000002                  BIS HO.VEC(R0),R4               ;ADD VECTOR ADDRESS
027000  020463  000004                  CMP R4,C.VEC(R3)                ;COMPARE WITH CONTROLLER TABLE
027004  001004                          BNE 1$
027006  026063  000006  000006        CMP HO.BST(R0),C.BST(R3)        ;COMPARE BURST RATES
027014  001402                          BEQ NXTTAB
027016  000137  027434          1$:     JMP CTABER                      ;FATAL ERROR IF NOT SAME
                                        
                                        ;GET NEXT P-TABLE
                                        
027022  005202                  NXTTAB: INC R2                          ;INCREMENT LOGICAL UNIT NUMBER
027024  023702  002012                  CMP L$UNIT,R2                   ;CHECK IF GOT ALL TABLES
027030  003305                          BGT INIT4                       ;IF NOT, GO BACK FOR NEXT
                                        
027032  012701  000001                  MOV #1,R1                       ;ALLOCATE SPACE FOR ZERO END WORD
027036  004737  012722                  CALL ALOCM                      ;AFTER CONTROLLER TABLES
```

```
                                ;NOW BUILD DRIVE TABLES

                                        CLR R5                  ;CLEAR CUSTOMER DATA FLAG
                                        CLR R2                  ;LOGICAL UNIT NUMBER IN R2
                                INIT8:  GPHARD R2,R0            ;GET POINTER TO A P-TABLE
                                                                        MOV     R2,R0
                                                                        TRAP    C$GPHRD
                                        BNCOMPLETE INIT14       ;IF NOT AVAILABLE, GO GET NEXT
                                                                        BCC     INIT14

                                ;FIND CONTROLLER TABLE

                                        MOV CTABS,R3            ;GET ADDRESS OF CONTROLLER TABLES
                                INIT10: CMP (R0),(R3)           ;CHECK IF SAME UNIBUS ADDRESS
                                        BEQ INIT11              ;BRANCH IF TABLE FOUND
                                        ADD #C.SIZE,R3          ;MOVE TO NEXT TABLE
                                        BR INIT10
```

INITIALIZE SECTION

```
                                        ;BUILD DRIVE TABLE
                                        
027072  012701  000103          INIT11: MOV #D.SIZE/2,R1                ;GET SIZE OF DRIVE TABLE
027076  004737  012722                  CALL ALOCM                      ;ALLOCATE SPACE FROM FREE MEMORY
                                ;       R0 POINTS TO P-TABLE
                                ;       R1 POINTS TO DRIVE TABLE
                                ;       R3 POINTS TO CONTROLLER TABLE
                                ;       R2 IS LOGICAL UNIT NUMBER
027102  062703  000020                  ADD #C.DR0,R3                   ;BUILD POINTER TO C.DR ENTRY IN CONTROLLER TABLE
027106  012704  000010                  MOV #8.,R4                      ;GET MAX COUNT OF DRIVES ON ONE CONTROLLER
027112  005713                  INIT12: TST (R3)                        ;CHECK IF ENTRY CONTAINS POINTER TO DRIVE TABLE
027114  001411                          BEQ INIT13
027116  026033  000010                  CMP HO.LDR(R0),@(R3)+           ;CHECK DRIVE NUMBER IN DRIVE TABLE
027122  001002                          BNE 1$
027124  000137  027446                  JMP MLDRER                      ;IF SAME, TWO P-TABLES POINT TO SAME DRIVE
027130  005304                  1$:     DEC R4                          ;COUNT DRIVES
027132  001367                          BNE INIT12                      ;IF EIGHT DRIVE TABLES EXIST,
027134  000137  027460                  JMP TOOMER                      ;  THEN REPORT ERROR
027140  010113                  INIT13: MOV R1,(R3)                     ;LOAD DRIVE TABLE POINTER
027142  016021  000010                  MOV HO.LDR(R0),(R1)+            ;LOAD DRIVE NUMBER
027146  010221                          MOV R2,(R1)+                    ;LOAD UNIT NUMBER
027150  016011  000012                  MOV HO.PRM(R0),(R1)             ;GET TEST AREA BIT
027154  051105                          BIS (R1),R5                     ;SAVE 'OR' OF BIT FROM ALL DRIVES
027156  005111                          COM (R1)                        ;COMPLIMENT IT
027160                                  AND HM.CYL,(R1)
027160  042711  157777                      BIC #^C<HM.CYL>,(R1)
027164  052721  011012                  BIS #DDEF,(R1)+                 ;LOAD DEFAULT PARAMETER BITS
027170  012703  000100                  MOV #<D.SIZE/2-3>,R3            ;CLEAR REST OF TABLE
027174  005021                  INIT3L: CLR (R1)+
027176  005303                          DEC R3
027200  003375                          BGT INIT3L
027202  012761  177777  177754          MOV #-1,<D.ECYL+2-D.SIZE>(R1)   ;MARK CYLINDERS AT TEST ALL
```

INITIALIZE SECTION

```
                                        ;GO TO NEXT DRIVE TABLE
                                        ;
027210  005202                  INIT14: INC R2                          ;INCREMENT LOGICAL UNIT NUMBER
027212  023702  002012                  CMP L$UNIT,R2                   ;CHECK IF GOT ALL TABLES
027216  003313                          BGT INIT8                       ;IF NOT, GET NEXT TABLE
                                        ;
                                        ;IF ANY DRIVE SELECTED FOR EXERCISE IN CUSTOMER DATA AREA
                                        ;GIVE WARNING
                                        ;
027220  032705  020000                  BIT #HM.CYL,R5                  ;CHECK IF BIT EVER SET
027224  001464                          BEQ INIT15                      ;BYPASS IF NOT
027226                                  PRINTF #INITWA                  ;PRINT WARNING HEADER
027226  012746  007053                                                          MOV     #INITWA,-(SP)
027232  012746  000001                                                          MOV     #1,-(SP)
027236  010600                                                                  MOV     SP,R0
027240  104417                                                                  TRAP    C$PNTF
027242  062706  000004                                                          ADD     #4,SP
027246  013705  002204                  MOV CTABS,R5                    ;GET FIRST CONTROLLER TABLE
027252  010504          INITW1: MOV R5,R4                               ;GET ADDRESS OF POINTER TO DRIVE TABLE
027254  062704  000020                  ADD #C.DRO,R4
027260  012701  000010                  MOV #8.,R1                      ;GET COUNT OF DRIVE TABLES
027264  012403          INITW2: MOV (R4)+,R3                            ;GET ADDRESS OF DRIVE TABLE
027266  001422                          BEQ INITW4
027270  032763  020000  000004          BIT #D.DCY,D.PRM(R3)            ;CHECK IF CUSTOMER DATA SELECTED
027276  001014                          BNE INITW3
027300                                  PRINTF #INITWB,D.UNIT(R3),(R5),(R3)     ;PRINT NUMBERS
027300  011346                                                                  MOV     (R3),-(SP)
027302  011546                                                                  MOV     (R5),-(SP)
027304  016346  000002                                                          MOV     D.UNIT(R3),-(SP)
027310  012746  007165                                                          MOV     #INITWB,-(SP)
027314  012746  000004                                                          MOV     #4,-(SP)
027320  010600                                                                  MOV     SP,R0
027322  104417                                                                  TRAP    C$PNTF
027324  062706  000012                                                          ADD     #12,SP
027330  005301          INITW3: DEC R1                                  ;COUNT THE DRIVE TABLES
027332  001354                  BNE INITW2                              ;LOOK AT ALL OF THEM
027334  062705  000046          INITW4: ADD #C.SIZE,R5                  ;MOVE TO NEXT CONTROLLER TABLE
027340  005715                  TST (R5)                                ;SEE IF ANOTHER TABLE AND
027342  001343                  BNE INITW1                              ;  LOOK AT IT
                                        ;
                                        ;GET CONFIRMATION TO PROCEED
                                        ;
027344                                  MANUAL                          ;CHECK IF MANUAL INTERVENTION ALLOWED
027344  104450                                                                  TRAP    C$MANI
027346                                  BNCOMPLETE INIT15               ;BRANCH IF ALLOWED
027346  103013                                                                  BCC     INIT15
027350                                  GMANIL INITWC,TEMP,1,NO         ;ASK OPERATOR
027350  104443                                                                  TRAP    C$GMAN
027352  000404                                                                  BR      10000$
027354  002260                                                                  .WORD   TEMP
027356  000120                                                                  .WORD   T$CODE
027360  005171                                                                  .WORD   INITWC
027362  000001                                                                  .WORD   1
027364                                                                  10000$:
027364  032737  000001  002260          BIT #1,TEMP                     ;LOOK AT RESPONSE
027372  001001                          BNE INIT15                      ;BRANCH IF YES WAS ANSWER
027374                                  DOCLN                           ;ABORT PROGRAM
```

INITIALIZE SECTION

```
     027374  104444                                                          TRAP    C$DCLN
 36
 37                                 ;SAVE CURRENT PARAMETERS TO FREE MEMORY SO EACH TEST CAN USE ALL OF IT
 38
 39 027376  013737  002174  002200  INIT15: MOV FFREE,FMEM                  ;SAVE START ADDRESS
 40 027404  013737  002176  002202          MOV FSIZE,FMEMS                 ;SAVE SIZE
 41
 42 027412                          INITXX: SETPRI  #PRI00                  ; SET RUNNING PRIORITY TO ZERO
    027412  012700  000000                                                  MOV     #PRI00,R0
    027416  104441                                                          TRAP    C$SPRI
 43 027420  005037  002412                  CLR DLL                         ;ERASE DOWNLINE LOAD DATA
 44 027424  004737  023716                  CALL CLOSEF                     ;MAKE SURE DATA FILE IS CLOSED
 45 027430                          KPRI:   EXIT iNIT
    027430  104432                                                          TRAP    C$EXIT
    027432  000040                                                          .WORD   L10027-.
```

```
      1                                 ;DIFFERENT VECTORS, BR LEVELS OR BURST RATES FOR ONE CONTROLLER
      2 027434                          CTABER. ERRSF 1,,CTABE
        027434  104454                                                          TRAP    C$ERSF
        027436  000001                                                          .WORD   1
        027440  000000                                                          .WORD   0
        027442  012612                                                          .WORD   CTABE
      3 027444                                DOCLN
        027444  104444                                                          TRAP    C$DCLN
      4
      5                                 ;TWO P-TABLES FOR SAME DRIVE
      6 027446                          MLDRER: ERRSF 1,MLDREM
        027446  104454                                                          TRAP    C$ERSF
        027450  000001                                                          .WORD   1
        027452  003370                                                          .WORD   MLDREM
        027454  000000                                                          .WORD   0
      7 027456                                DOCLN
        027456  104444                                                          TRAP    C$DCLN
      8
      9                                 ;MORE THAN EIGHT DRIVES SELECTED ON ONE CONTROLLER
     10
     11 027460                          TOOMER: ERRSF 1,TOOMEM
        027460  104454                                                          TRAP    C$ERSF
        027462  000001                                                          .WORD   1
        027464  003431                                                          .WORD   TOOMEM
        027466  000000                                                          .WORD   0
     12 027470                                DOCLN
        027470  104444                                                          TRAP    C$DCLN
     13
     14
     15 027472                                ENDINIT
        027472                                                          L10027:
        027472  104411                                                          TRAP    C$INIT
```

```
                                .SBTTL  AUTODROP SECTION

                                ;++
                                ; THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF
                                ; THE ''ADR'' FLAG WAS SET.  THE UNIT(S) UNDER TEST ARE CHECKED TO
                                ; SEE IF THEY WILL RESPOND.  THOSE THAT DON'T ARE IMMEDIATELY
                                ; DROPPED FROM TESTING.
                                ;--

 027474                                 BGNAUTO
 027474                                                                 L$AUTO::

 027474                                 ENDAUTO
 027474                                                                 L10030:
 027474  104461                                                                 TRAP    C$AUTO
```

CLEANUP CODING SECTION

```
      1                                 .SBTTL  CLEANUP CODING SECTION
      2
      3                                 ;++
      4                                 ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
      5                                 ; AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.
      6                                 ;--
      7
      8 027476                                  BGNCLN
        027476                                                                  L$CLEAN::
      9
     18 027476  004737  023716                  CALL CLOSEF                 ;CLOSE DATA FILE
     30 027502                                  ENDCLN
        027502                                                                  L10031:
        027502  104412                                                                  TRAP    C$CLEAN
     31 027504                                  ENDMOD
     32                                 .SBTTL
```

```
                                ;.TITLE HARDWARE TESTS   (5)

                                .SBTTL  TEST 1: UNIBUS ADDRESSING TEST

027504                                  BGNMOD

027504                                  BGNTST
027504                                                                  T1::
027504  012701  000001                    MOV     #1,R1                   ; INITIALIZE TEST PARAMETERS
027510  004737  013006                    CALL    TINIT
027514  013737  002204  002210            MOV     CTABS,TSTTAB            ; GET ADDRESS OF FIRST CONTROLLER TABLE
027522  013705  002210          T1NEXT:   MOV     TSTTAB,R5               ; GET CONTROLLER TABLE ADDRESS
027526  016537  000002  002074            MOV     C.UNIT(R5),L$LUN        ; CHECK IF UNIT AVAILABLE FOR TESTING
027534  100504                            BMI     T1SKIP                  ; SKIP TESTING IF NOT AVAILABLE
027536                                          ASSUME CT.AVL EQ BIT15

027536  004737  024414                    CALL    RESET                   ; RESET ALL UNITS
027542                          BGNSUB: 1
027542                                                                  T1.1:
027542  104402                                                                  TRAP    C$BSUB
027544  005037  002366                    CLR NXMAD                       ;CLEAR MEMORY ERROR FLAG
027550                                  SETVEC #4,#NXMI,#PRI07
027550  012746  000340                                                          MOV     #PRI07,-(SP)
027554  012746  022640                                                          MOV     #NXMI,-(SP)
027560  012746  000004                                                          MOV     #4,-(SP)
027564  012746  000003                                                          MOV     #3,-(SP)
027570  104437                                                                  TRAP    C$SVEC
027572  062706  000010                                                          ADD     #10,SP
027576  011504                    MOV (R5),R4                             ;GET ADDRESS OF UDAIP REGISTER
027600  005714                    TST (R4)                                ;READ UDAIP
027602  005764  000002            TST 2(R4)                               ;READ UDASA
027606                            CLRVEC  #4                              ;GIVE UP VECTOR
027606  012700  000004                                                          MOV     #4,R0
027612  104436                                                                  TRAP    C$CVEC
027614  005737  002366                    TST NXMAD                       ;CHECK FLAG
027620  001405                            BEQ T1GOOD
027622                                  ERRDF 21,0,INTR4
027622  104455                                                                  TRAP    C$ERDF
027624  000025                                                                  .WORD   21
027626  000000                                                                  .WORD   0
027630  012154                                                                  .WORD   INTR4
027632  000445                    BR T1SKIP                               ;DON'T TEST ANY MORE
027634                          T1GOOD:
027634                          ENDSUB
027634                                                                  L10033:
027634  104403                                                                  TRAP    C$ESUB
027636                          BGNSUB: 2
027636                                                                  T1.2:
027636  104402                                                                  TRAP    C$BSUB
027640  004737  027764                    CALL    PORTST                  ; DIAGNOSTIC LOOP MODE TESTING
027644                          ENDSUB
027644                                                                  L10034:
027644  104403                                                                  TRAP    C$ESUB
027646                          BGNSUB: 3
027646                                                                  T1.3:
027646  104402                                                                  TRAP    C$BSUB
027650  004737  030234                    CALL    INTEST                  ; TEST INTERRUPTS
```

```
     46 027654                          ENDSUB
        027654
        027654  104403                                                                  L10035:
                                                                                                TRAP    C$ESUB
     47 027656                          BGNSUB: 4
        027656                                                                          T1.4:
        027656  104402                                                                          TRAP    C$BSUB
     48 027660  005004                          CLR     R4                  ; INITIALIZE UDA WITH SMALLEST
     49 027662  004737  022742                  CALL    UDAINT              ; RING BUFFER AND INTERRUPTS DISABLED
     50 027666                          ENDSUB
        027666
        027666  104403                                                                  L10036:
                                                                                                TRAP    C$ESUB
     51 027670                          BGNSUB: 5
        027670                                                                          T1.5:
        027670  104402                                                                          TRAP    C$BSUB
     52 027672  012704  132000                  MOV     #132000,R4          ;INITIALIZE UDA WITH RING BUFFER LARGE
     53 027676  004737  022742                  CALL    UDAINT              ; ENOUGH TO COVER NORMAL HOST COMM AREA
     54                                                                     ; PACKET AND BUFFER SPACE (A 5 IN MES
     55                                                                     ; LENGTH AND A 5 IN CMD LENGTH)
     56 027702                          ENDSUB
        027702                                                                          L10037:
        027702  104403                                                                          TRAP    C$ESUB
     57 027704                          BGNSUB: 6
        027704                                                                          T1.6:
        027704  104402                                                                          TRAP    C$BSUB
     58 027706                                  PUSH    FFREE               ; SAVE FREE MEMORY PARAMETERS
        027706  013746  002174                          MOV FFREE,-(SP)
     59 027712                                  PUSH    FSIZE
        027712  013746  002176                          MOV FSIZE,-(SP)
     60 027716  012701  000001                  MOV     #1,R1               ; RUN DM PROGRAM IN
     61 027722  004737  013134                  CALL    RUNDM               ; ONE CONTROLLER ONLY
     62 027726  001402                          BEQ 1$
     63 027730  004737  013224                  CALL RESPDM
     64 027734                          1$:     POP     FSIZE
        027734  012637  002176                          MOV (SP)+,FSIZE
     65 027740                                  POP     FFREE
        027740  012637  002174                          MOV (SP)+,FFREE
     66 027744                          ENDSUB
        027744                                                                  L10040:
        027744  104403                                                                  TRAP    C$ESUB
```

TEST 1: UNIBUS ADDRESSING TEST

```
      1 027746  062737  000046  002210  T1SKIP: ADD     #C.SIZE,TSTTAB          ; MOVE TO NEXT CONTROLLER
      2 027754  005777  152230                  TST     @TSTTAB                 ; CHECK IF ANOTHER CONTROLLER TABLE
      3 027760  001260                           BNE     T1NEXT
      4
      5 027762                           T1ERR:
      6 027762                                   ENDTST
        027762                                                          L10032:
        027762  104401                                                         TRAP    CSETST
```

```
                                PORTST:
                                ;
                                ;       TEST THE DIAGNOSTIC LOOP MODE OF ALL UDA'S ON THE SYSTEM
                                ;
027764  011504                  MOV     (R5),R4                 ; R4 POINTS TO UDAIP REGISTER
027766                          ASSUME C.UADR EQ 0
027766  005014                  CLR     (R4)                    ; INITIALIZE THE UDA
027770  012737  004000  023652  MOV     #SA.S1,UDARSD           ; LOOK FOR STEP 1
027776  004737  023514          CALL    UDARSP                  ; WAIT FOR RESPONSE
030002  103453                  BCS     3$                      ; IF ERROR, BRANCH
030004  016437  000002  030232  MOV     2(R4),WCHNGD            ; MOVE OLD PORT CONTENTS TO STORAGE
030012  012764  140000  000002  MOV     #140000,2(R4)           ; INITIALIZE FOR PORT WRAP
030020  004737  030136          CALL    WCHNG                   ; WAIT FOR THE PORT TO CHANGE
030024  001442                  BEQ     3$                      ; IF ERROR, BRANCH
030026  022764  140000  000002  CMP     #140000,2(R4)           ; COMPARE WITH DATA WRITTEN
030034  001405                  BEQ     4$                      ; IF NO ERROR, BRANCH
030036                          ERRDF   58,LOOP02,LOOPA         ; REPORT ERROR
030036  104455                                                          TRAP    C$ERDF
030040  000072                                                          .WORD   58
030042  005012                                                          .WORD   LOOP02
030044  012534                                                          .WORD   LOOPA
030046  000431                  BR      3$                      ; EXIT, REPORTING ERROR TO CALLING ROUTINE
030050  012702  000001  4$:     MOV     #1,R2                   ; SET UP FOR SHIFTING '1'
030054  012703  000020          MOV     #16.,R3                 ; SET UP LOOP COUNT
030060  016437  000002  030232  1$:     MOV     2(R4),WCHNGD    ; SAVE OLD PORT CONTENTS
030066  010264  000002          MOV     R2,2(R4)                ; WRITE PATTERN TO UDASA FOR LOOP
030072  004737  030136          CALL    WCHNG                   ; WAIT FOR UDASA TO CHANGE
030076  001415                  BEQ     3$                      ; IF ERROR, BRANCH
030100  020264  000002          CMP     R2,2(R4)                ; COMPARE R0 WITH WHAT WAS ECHOED
030104  001405                  BEQ     2$                      ; IF MATCH, BRANCH
030106                          ERRDF   57,LOOP02,LOOP          ; REPORT ERROR
030106  104455                                                          TRAP    C$ERDF
030110  000071                                                          .WORD   57
030112  005012                                                          .WORD   LOOP02
030114  012504                                                          .WORD   LOOP
030116  000405                  BR      3$                      ; BRANCH
030120  006302          2$:     ASL     R2                      ; MOVE THE SHIFTING ONE LEFT BY 1
030122  005303                  DEC     R3                      ; DECREMENT COUNT
030124  001355                  BNE     1$                      ; IF LOOP INCOMPLETE, BRANCH
030126  000244                  CLZ                             ; FLAG AS NO ERRORS
030130  000207                  RETURN                          ; RETURN TO TEST 1 CODE
030132  000264          3$:     SEZ                             ; FLAG AS ERROR OCCURRED
030134  000207                  RETURN                          ; RETURN TO TEST 1 CODE
```

```
                                        ;WCHNG
                                        ;
                                        ;       WAIT UNTIL UDASA CHANGES FROM WHAT IS IN WCHNGD
                                        
030136  012700  000012          WCHNG:  MOV #10.,R0                     ;SET TIMEOUT FOR 10 SECONDS
030142  010501                          MOV R5,R1                       ;POINT TO CONTROLLER TABL.
030144  062701  000040                  ADD #C.TO,R1
030150  004737  022660                  CALL SETTO
030154  026437  000002  030232  1$:     CMP 2(R4),WCHNGD                ;SEE IF CHANGED
030162  001022                          BNE 2$
030164                                  BREAK
030164  104422                                                                  TRAP    C$BRK
030166  005737  002352                  TST KW.CSR                      ;SEE IF CLOCK ON SYSTEM
030172  001770                          BEQ 1$
030174  023765  002364  000042          CMP KW.EL+2,C.TOH(R5)           ;CHECK IF TIME OUT OCCURRED
030202  101005                          BHI 3$
030204  001363                          BNE 1$
030206  023765  002362  000040          CMP KW.EL,C.TO(R5)
030214  103757                          BLO 1$
030216                          3$:     ERRDF   56,LOOP01               ; REPORT ERROR
030216  104455                                                                  TRAP    C$ERDF
030220  000070                                                                  .WORD   56
030222  004715                                                                  .WORD   LOOP01
030224  000000                                                                  .WORD   0
030226  000264                          SEZ                             ; FLAG AS ERROR
030230  000207                  2$:     RETURN                          ; RETURN TO CALLING PROGRAM

030232                          WCHNGD: .BLKW   1                       ; OLD PORT CONTENTS
```

TEST 1: UNIBUS ADDRESSING TEST

```
      1 030234                          INTEST:
      2                                 ;
      3                                 ;       TEST THE INTERRUPTS VECTOR AND BR LEVEL
      4
      5 030234  011504                          MOV     (R5),R4                 ; R4 POINTS TO UDAIP REGISTER
      6 030236                                  ASSUME C.UADR EQ 0
      7 030236  016503  000004                  MOV     C.VEC(R5),R3            ; GET VECTOR AND BRANCH LEVEL
      8 030242  010302                          MOV     R3,R2                   ; COPY TO R2 FOR BR LEVEL
      9 030244  042703  177000                  BIC     #^CCT.VEC,R3            ; CLEAR UNUSED VECTOR BITS
     10 030250  042702  170777                  BIC     #^CCT.BRL,R2            ; CLEAR UNUSED BRANCH LEVEL BITS
     11 030254  012701  000011                  MOV     #9.,R1                  ; SET UP TO SHIFT BR LEVEL
     12 030260  006202                  1$:     ASR     R2                      ; SHIFT BY ONE BIT
     13 030262  005301                          DEC     R1                      ; COUNT SHIFTS
     14 030264  001375                          BNE     1$                      ; IF INCOMPLETE, BRANCH
     15 030266  010237  030706                  MOV     R2,BRLEV                ; SAVE THE BRANCH LEVEL
     16 030272                                  PRINTF  #INTST0,C.UADR(R5),R3   ; PRINT BEGINNING OF INTERRUPT MESSAGE
        030272  010346                                                                  MOV     R3,-(SP)
        030274  016546  000000                                                          MOV     C.UADR(R5),-(SP)
        030300  012746  006672                                                          MOV     #INTST0,-(SP)
        030304  012746  000003                                                          MOV     #3,-(SP)
        030310  010600                                                                  MOV     SP,R0
        030312  104417                                                                  TRAP    C$PNTF
        030314  062706  000010                                                          ADD     #10,SP
     17 030320                                  SETVEC  R3,#INTSRV,#PRI00       ; SET UP INTERRUPT ROUTINE
        030320  012746  000000                                                          MOV     #PRI00,-(SP)
        030324  012746  024406                                                          MOV     #INTSRV,-(SP)
        030330  010346                                                                  MOV     R3,-(SP)
        030332  012746  000003                                                          MOV     #3,-(SP)
        030336  104437                                                                  TRAP    C$SVEC
        030340  062706  000010                                                          ADD     #10,SP
     18 030344                                  SETPRI  #PRI00                  ; SET PRIORITY TO 0 TO CHECK INTERRUPTS
        030344  012700  000000                                                          MOV     #PRI00,R0
        030350  104441                                                                  TRAP    C$SPRI
     19 030352  006203                          ASR     R3                      ; DIVIDE VECTOR BY 4 FOR UDA INITIALIZATION
     20 030354  006203                          ASR     R3                      ; DIVIDE VECTOR BY 4 FOR UDA INITIALIZATION
     21 030356  052703  104600                  BIS     #104600,R3              ; SET OTHER BITS FOR UDA INITIALIZATION
     22 030362  005037  002236                  CLR     INTRCV                  ; FLAG AS NO INTERRUPTS RECEIVED
     23 030366  005014                          CLR     (R4)                    ; INIT UDA
     24 030370  012737  004000  023652          MOV     #SA.S1,UDARSD           ; LOOK FOR STEP 1 COMPLETION
     25 030376  004737  023514                  CALL    UDARSP                  ; WAIT FOR COMPLETION
     26 030402  010364  000002                  MOV     R3,2(R4)                ; MOVE STEP 1 DATA TO UDA
     27 030406  012700  000012                  MOV #10.,R0                     ;SET UP TIMEOUT OF 10 SECONDS
     28 030412  010501                          MOV R5,R1
     29 030414  062701  000040                  ADD #C.TO,R1                    ;POINT TO CONTROLLER TABLE
     30 030420  004737  022660                  CALL SETTO
     31 030424  005737  002236          9$:     TST     INTRCV                  ; SEE IF INTERRUPTED
     32 030430  001022                          BNE     11$                     ; IF SO, EVERYTHING'S OK, SO BRANCH
     33 030432                                  BREAK
        030432  104422                                                                 TRAP    C$BRK
     34 030434  005737  002352                  TST KW.CSR                      ;SEE IF CLOCK ON SYSTEM
     35 030440  001771                          BEQ 9$
     36 030442  023765  002364  000042          CMP KW.EL+2,C.TOH(R5)           ;SEE IF TIME ELAPSED
     37 030450  101005                          BHI 3$
     38 030452  001364                          BNE 9$
     39 030454  023765  002362  000040          CMP KW.EL,C.TO(R5)
     40 030462  103760                          BLO 9$
     41 030464                          3$:     ERRDF   59,INTST4               ; REPORT NO INTERRUPTS ERROR
```

```
                030464  104455                                                          TRAP    C$ERDF
                030466  000073                                                          .WORD   59
                030470  005124                                                          .WORD   INTST4
                030472  000000                                                          .WORD   0
     42         030474  000474                      BR      6$                  ; BRANCH
     43         030476  005037  002236      11$:    CLR     INTRCV              ; FLAG AS NO INTERRUPTS RECEIVED
     44         030502                      SETPRI  #PRI07                      ; SET PRIORITY AS HIGHEST PRIORITY
                030502  012700  000340                                                  MOV     #PRI07,R0
                030506  104441                                                          TRAP    C$SPRI
     45         030510  005064  000002              CLR     2(R4)               ; WRITE SECOND STEP TO UDA
     46         030514  012702  000144              MOV     #100.,R2            ; SET UP DELAY SO WE KNOW WE'RE INTERRUPTED
     47         030520  005302              12$:    DEC     R2                  ; DECREMENT COUNT
     48         030522  001376                      BNE     12$                 ; IF INCOMPLETE, BRANCH
     49         030524  012701  000007              MOV     #7.,R1              ; R1 IS PROCESS PRIORITY LEVEL
     50         030530                      2$:     PUSH    R1                  ; SAVE PRIORITY
                030530  010146                              MOV R1,-(SP)
     51         030532  012702  000005              MOV     #5.,R2              ; SET UP FOR SHIFTING PRIORITY
     52         030536  006301              10$:    ASL     R1                  ; SHIFT PRIORITY
     53         030540  005302                      DEC     R2                  ; DECREMENT SHIFT COUNT
     54         030542  001375                      BNE     10$                 ; IF INCOMPLETE, BRANCH
     55         030544                      SETPRI  R1                          ; SET RUNNING PRIORITY TO R1
                030544  010100                                                          MOV     R1,R0
                030546  104441                                                          TRAP    C$SPRI
     56         030550                      POP     R1                  ; RESTORE R1
                030550  012601                              MOV (SP)+,R1
     57         030552  005737  002236              TST     INTRCV              ; SEE IF INTERRUPT RECEIVED
     58         030556  001007                      BNE     4$                  ; IF SO, BRANCH
     59         030560  005301                      DEC     R1                  ; DECREMENT PRIORITY LEVEL
     60         030562  100362                      BPL     2$                  ; IF ALL LEVELS UNTESTED, BRANCH
     61         030564                      ERRDF   59,INTST4                   ; REPORT NO INTERRUPTS ERROR
                030564  104455                                                          TRAP    C$ERDF
                030566  000073                                                          .WORD   59
                030570  005124                                                          .WORD   INTST4
                030572  000000                                                          .WORD   0
     62         030574  000434                      BR      6$                  ; BRANCH
     63         030576                      4$:     SETPRI  #PRI00              ; SET RUNNING PRIORITY TO 0
                030576  012700  000000                                                  MOV     #PRI00,R0
                030602  104441                                                          TRAP    C$SPRI
     64         030604  005201                      INC     R1                  ; SO PRIORITY = BR LEVEL
     65         030606  023701  030706              CMP     BRLEV,R1            ; SEE IF BR LEVEL MATCHES PRIORITY
     66         030612  001405                      BEQ     5$                  ; IF SO, BRANCH
     67         030614                      ERRDF   60,INTST2,INTERR            ; REPORT ERROR
                030614  104455                                                          TRAP    C$ERDF
                030616  000074                                                          .WORD   60
                030620  005101                                                          .WORD   INTST2
                030622  012566                                                          .WORD   INTERR
     68         030624  000420                      BR      6$                  ; BRANCH
     69         030626                      5$:     PRINTF  #INTST1             ; PRINT TESTING COMPLETED
                030626  012746  006771                                                  MOV     #INTST1,-(SP)
                030632  012746  000001                                                  MOV     #1,-(SP)
                030636  010600                                                          MOV     SP,R0
                030640  104417                                                          TRAP    C$PNTF
                030642  062706  000004                                                  ADD     #4,SP
     70         030646  016503  000004              MOV     C.VEC(R5),R3        ; GET VECTOR ADDRESS
     71         030652  042703  177000              BIC     #^CCT.VEC,R3        ; CLEAR UNUSED BITS
     72         030656                      CLRVEC  R3                          ; CLEAR VECTOR
                030656  010300                                                          MOV     R3,R0
```

TEST 1: UNIBUS ADDRESSING TEST

```
                030660  104436                                                          TRAP    C$CVEC
     73         030662  000244                          CLZ                             ; FLAG AS NO ERRORS
     74         030664  000207                          RETURN                          ; RETURN TO TEST 1
     75         030666  016503  000004          6$:     MOV     C.VEC(R5),R3            ; GET VECTOR ADDRESS
     76         030672  042703  177000                  BIC     #^CCT.VEC,R3            ; CLEAR UNUSED BITS
     77         030676                                  CLRVEC  R3                      ; CLEAR VECTOR
                030676  010300                                                          MOV     R3,R0
                030700  104436                                                          TRAP    C$CVEC
     78         030702  000264                          SEZ                             ; FLAG AS ERROR OCCURRED
     79         030704  000207                          RETURN                          ; RETURN TO TEST 1
     80
     81
     82
     83
     84         030706                          BRLEV:  .BLKW   1                       ; WORD FOR BRANCH LEVEL STORAGE
```

TEST 2: DISK RESIDENT DIAGNOSTIC TEST

```
                                .SBTTL  TEST 2: DISK RESIDENT DIAGNOSTIC TEST
                                
030710                                  BGNTST
030710                                                                          T2::
                                
030710  012701  000002                  MOV #2,R1                       ;INIT TEST PARAMETERS
030714  004737  013006                  CALL TINIT
                                
030720  013737  002204  002210          MOV CTABS,TSTTAB                ;GET POINTER TO FIRST CONTROLLER TABLE
                                
030726  004737  024414          T2NEXT: CALL RESET                      ;RESET ALL UNITS
030732                                  PUSH FFREE                      ;SAVE FREE MEMORY PARAMETERS
030732  013746  002174                          MOV FFREE,-(SP)
030736                                  PUSH FSIZE
030736  013746  002176                          MOV FSIZE,-(SP)
030742  012701  000001                  MOV #1,R1                       ;RUN DM PROGRAM IN
030746  004737  013134                  CALL RUNDM                      ; ONE CONTROLLER ONLY
030752  001402                          BEQ 1$
030754  004737  013224                  CALL RESPDM
030760                          1$:     POP FSIZE
030760  012637  002176                          MOV (SP)+,FSIZE
030764                                  POP FFREE
030764  012637  002174                          MOV (SP)+,FFREE
                                
030770  062737  000046  002210          ADD #C.SIZE,TSTTAB              ;MOVE TO NEXT CONTROLLER
031002  005777  151206                  TST @TSTTAB                     ;CHECK IF ANY MORE CONTROLLER TABLES
031002  001351                          BNE T2NEXT
                                
031004                                  ENDTST
031004                                                                  L10041:
031004  104401                                                                  TRAP    C$ETST
```

TEST 3: DISK FUNCTION TEST

```
                                .SBTTL  TEST 3: DISK FUNCTION TEST
                                
031006                                  BGNTST
031006                                                                  T3::
                                
031006  012701  000003                  MOV #3,R1               ;INITIALIZE TEST PARAMETERS
031012  004737  013006                  CALL TINIT
                                
031016  013737  002204  002210          MOV CTABS,TSTTAB        ;GET FIRST TABLE ADDRESS
031024  013701  002206                  MOV CTRLRS,R1           ;RUN DM PROGRAM ON ALL CONTROLLERS
031030  004737  013134                  CALL RUNDM              ;  AT ONCE
031034  001402                          BEQ 1$
031036  004737  013224                  CALL RESPDM
031042                          1$:
031042                                  ENDTST
031042                                                          L10042:
031042  104401                                                          TRAP    C$ETST
```

```
                                .SBTTL  TEST 4: DISK EXERCISER

                                        BGNTST
                                                                                T4::
                                        CMP #4,TNUM             ;CHECK IF TEST 4 WAS IN PROGRESS
                                        BNE T4STRT              ;BRANCH IF NOT
                                        CMP #ICONT,IFLAGS       ;CHECK IF HERE BY CONTINUE COMMAND
                                        BNE T4STRT              ;BRANCH IF NOT
                                        CLR IFLAGS              ;CLEAR FLAGS FOR NEXT TIME HERE
                                        MOV LBUFS,R4            ;GET LOG BUFFER POINTER
                                        BEQ LOGCHK              ;   IF ZERO, NONE EXISTS
                                        PRINTF #LOGM1           ;INTRODUCE ERROR LOG
                                                                        MOV     #LOGM1,-(SP)
                                                                        MOV     #1,-(SP)
                                                                        MOV     SP,R0
                                                                        TRAP    C$PNTF
                                                                        ADD     #4,SP
                                        CLR LBUFS               ;CLEAR START ADDRESS TO ERASE BUFFER
LOGOUT:                 MOV (R4)+,R5    ;GET CONTROLLER TABLE ADDRESS
                                        CALL PNTERR             ;PRINT ERROR REPORT
                                        ADD #<HC.BSZ-2>,R4      ;BUMP POINTER TO NEXT ENTRY
                                        CMP R4,LBUFN            ;CHECK IF AT END
                                        BLO LOGOUT              ;PRINT ALL ENTRIES
                                        PRINTF #LOGM2
                                                                        MOV     #LOGM2,-(SP)
                                                                        MOV     #1,-(SP)
                                                                        MOV     SP,R0
                                                                        TRAP    C$PNTF
                                                                        ADD     #4,SP
                                        BR T4CON

LOGCHK:                 BIT #SM.LOG,SFPTBL+S0.BIT       ;CHECK IF LOG ENABLED
                                        BEQ T4CON
                                        PRINTF #LOGM3           ;REPORT LOG EMPTY
                                                                        MOV     #LOGM3,-(SP)
                                                                        MOV     #1,-(SP)
                                                                        MOV     SP,R0
                                                                        TRAP    C$PNTF
                                                                        ADD     #4,SP
T4CON:                  TST URNING              ;CHECK IF ANY CONTROLLERS STILL RUNNING
                                        BEQ T4STRT              ;RESTART IF NOT
                                        CALL RESPDM             ;CONTINUE BY RESPONDING TO REQUESTS
                                        BR T4WAIT               ;END OF TEST WHEN DONE

                                ;START TEST 4

T4STRT:                 BIT #ISTRT+IREST,IFLAGS         ;HERE FROM OPERATOR COMMAND?
                                        BEQ T4RUN               ;RUN WITH PREVIOUS PARAMETERS IF NEW PASS
                                        BIT #SM.MAN,SFPTBL+S0.BIT       ;MANUAL INTERVENTION MODE?
                                        BEQ T4DEF               ;IF NOT, SET UP DEFAULT PARAMETERS
                                        MANUAL                  ;MANUAL INTERVENTION ALLOWED?
                                                                        TRAP    C$MANI
                                        BNCOMPLETE T4DEFW       ;IF NOT, GIVE WARNING
                                                                        BCC     T4DEFW

                                ;INPUT PARAMETERS
```

```
     40
     41 031254  005037  002234            CLR UCNT                              ;CLEAR COUNT OF UNITS USING PATTERN 16
     42 031260  013705  002204            MOV CTABS,R5                          ;GET FIRST CONTROLLER TABLE
     43 031264  012702  000010  T4PRM1:    MOV #8.,R2                            ;GET COUNT OF DRIVE TABLES
     44 031270  010504                     MOV R5,R4                             ;GET FIRST DRIVE TABLE POINTER
     45 031272  062704  000020            ADD #C.DRO,R4
     46 031276  012403            T4PRM2:  MOV (R4)+,R3                          ;GET DRIVE TABLE ADDRESS
     47 031300  001416                     BEQ T4PRM4                            ;GO TO NEXT CONTROLLER IF NONE
     48 031302  032763  100000  000002     BIT #DT.AVL,D.UNIT(R3)                ;SEE IF TO BE TESTED
     49 031310  001010                     BNE T4PRM3
     50 031312  004737  031636            CALL T4QUEST                          ;ASK QUESTIONS
     51 031316  022763  000020  000006     CMP #16.,D.PAT(R3)
     52 031324  001002                     BNE T4PRM3
     53 031326  005237  002234            INC UCNT
     54 031332  005302            T4PRM3:  DEC R2                                ;COUNT DRIVE TABLES
     55 031334  001360                     BNE T4PRM2                            ;GO LOOK AT NEXT
     56 031336  062705  000046            T4PRM4: ADD #C.SIZE,R5                 ;GO TO NEXT CONTROLLER
     57 031342  005715                     TST (R5)                              ;  IF THERE IS ONE
     58 031344  001347                     BNE T4PRM1
     59
     60                                 ;NOW GET DATA PATTERN 16 IF SELECTED BY ANY DRIVE
     61
     62 031346  005737  002234            TST UCNT                              ;CHECK IF ANY UNITS USING PAT 16
     63 031352  001475                     BEQ T4RUN                             ;IF NONE, DON'T ASK ANY MORE QUESTIONS
     64 031354                             GMANID T4DPC,PAT16C,D,-1,1,16.,YES    ;COUNT OF WORDS
        031354  104443                                                                      TRAP    C$GMAN
        031356  000406                                                                      BR      10000$
        031360  002310                                                                      .WORD   PAT16C
        031362  000052                                                                      .WORD   T$CODE
        031364  004472                                                                      .WORD   T4DPC
        031366  177777                                                                      .WORD   -1
        031370  000001                                                                      .WORD   T$LOLIM
        031372  000020                                                                      .WORD   T$HILIM
        031374                                                                   10000$:
     65 031374  013701  002310                  MOV PAT16C,R1                    ;GET COUNT OF WORDS
     66 031400  012704  002312                  MOV #PAT16W,R4                   ;GET ADDRESS OF STORAGE
     67 031404  011437  002260     T4PRM5:      MOV (R4),TEMP
     68 031410                                 GMANID T4DPD,TEMP,0,-1,1,-1,YES ;DATA WORD
        031410  104443                                                                   TRAP    C$GMAN
        031412  000406                                                                   BR      10001$
        031414  002260                                                                   .WORD   TEMP
        031416  000032                                                                   .WORD   T$CODE
        031420  004535                                                                   .WORD   T4DPD
        031422  177777                                                                   .WORD   -1
        031424  000001                                                                   .WORD   T$LOLIM
        031426  177777                                                                   .WORD   T$HILIM
        031430                                                                10001$:
     69 031430  013724  002260                  MOV TEMP,(R4)+
     70 031434  005301                          DEC R1                           ;COUNT THE WORDS
     71 031436  001362                          BNE T4PRM5
     72 031440  000442                          BR T4RUN
     73
     74                                 ;GIVE WARNING MANUAL INTERVENTION NOT ALLOWED
     75
     76 031442                          T4DEFW: PRINTF #T4WARN
        031442  012746  007212                                                         MOV     #T4WARN,-(SP)
        031446  012746  000001                                                         MOV     #1,-(SP)
```

```
                031452  010600                                                          MOV     SP,R0
                031454  104417                                                          TRAP    C$PNTF
                031456  062706  000004                                                  ADD     #4,SP
            77
            78                                  ;SET UP DEFAULT PARAMETERS
            79
            80  031462  013705  002204          T4DEF:  MOV CTABS,R5                    ;GET FIRST CONTROLLER TABLE
            81  031466  012702  000010          T4DEFA: MOV #8.,R2                      ;GET COUNT OF DRIVE TABLES
            82  031472  010504                          MOV R5,R4                       ;GET FIRST DRIVE TABLE POINTER
            83  031474  062704  000020                  ADD #C.DR0,R4
            84  031500  012403                  T4DEFB: MOV (R4)+,R3                    ;GET DRIVE TABLE ADDRESS
            85  031502  001415                          BEQ T4DEFE                      ;GO TO NEXT CONTROLLER IF NONE
            86  031504  062703  000004                  ADD #D.PRM,R3
            87  031510                                  AND D.DCY,(R3)                  ;INITIALIZE ALL PARAMETER BITS
                031510  042713  157777                          BIC #^C<D.DCY>,(R3)
            88  031514  052723  011012                  BIS #DDEF,(R3)+
            89  031520  012700  000067                  MOV #55.,R0
            90  031524  005023                  T4DEFC: CLR (R3)+
            91  031526  005300                          DEC R0
            92  031530  001375                          BNE T4DEFC
            93  031532  005302                  T4DEFD: DEC R2                          ;COUNT DRIVE TABLES
            94  031534  001361                          BNE T4DEFB                      ;GO LOOK AT NEXT
            95  031536  062705  000046          T4DEFE: ADD #C.SIZE,R5                  ;GO TO NEXT CONTROLLER
            96  031542  005715                          TST (R5)                        ;  IF THERE IS ONE
            97  031544  001350                          BNE T4DEFA
            98
            99                                  ;START TEST 4
           100
           101  031546  006137  002224          T4RUN:  ROL IFLAGS                      ;CLEAR FLAGS FOR NEXT TIME HERE
           102  031552                                  AND ISTRTH,IFLAGS               ;HOLD START FOR T4UPRM REQUEST
                031552  042737  177757  002224                  BIC #^C<ISTRTH>,IFLAGS
           103  031560  012701  000004                  MOV #4,R1                       ;INITIALIZE TEST PARAMETERS
           104  031564  004737  013006                  CALL TINIT
           105
           106  031570  013737  002204  002210          MOV CTABS,TSTTAB                ;GET FIRST TABLE ADDRESS
           107  031576  013701  002206                  MOV CTRLRS,R1                   ;RUN DM PROGRAM ON ALL CONTROLLERS
           108  031602  004737  013134                  CALL RUNDM                      ;  AT ONCE
           109  031606  001402                          BEQ T4WAIT
           110  031610  004737  013224                  CALL RESPDM
           111  031614  032737  001000  002160  T4WAIT: BIT #SM.LOG,SFPTBL+S0.BIT       ;CHECK IF LOG IS ENABLED
           112  031622  001402                          BEQ T4EXIT                      ;EXIT IF NOT
           113  031624                                  BREAK
                031624  104422                                                          TRAP    C$BRK
           114  031626  000772                          BR T4WAIT                       ;WAIT TILL STOPPED BY CONTROL C
           115
           116  031630                          T4EXIT: DORPT                           ;PRINT STATISTICS
                031630  104424                                                          TRAP    C$DRPT
           117  031632                                  EXIT TST
                031632  104432                                                          TRAP    C$EXIT
                031634  001732                                                          .WORD   L10043-.
```

```
                                ;ASK TEST 4 MANUAL INTERVENTION QUESTIONS
                                ;
                                ;INPUTS:
                                ;       R5 - POINTER TO CONTROLLER TABLE
                                ;       R3 - POINTER TO DRIVE TABLE
                                ;       R2 AND R4 MUST BE PRESERVED
                                ;OUTPUTS:
                                ;       DRIVE TABLE WITH NEW PARAMETERS
                                ;       R0 AND R1 CONTENTS DESTROYED
                                ;
031636                          T4QUEST:PUSH <R2,R4>
031636  010246                                  MOV R2,-(SP)
031640  010446                                  MOV R4,-(SP)
031642                                  PRINTF #T4QHED,D.UNIT(R3),(R5),(R3)     ;PRINT HEADER
031642  011346                                                                  MOV     (R3),-(SP)
031644  011546                                                                  MOV     (R5),-(SP)
031646  016346  000002                                                         MOV     D.UNIT(R3),-(SP)
031652  012746  007321                                                         MOV     #T4QHED,-(SP)
031656  012746  000004                                                         MOV     #4,-(SP)
031662  010600                                                                  MOV     SP,R0
031664  104417                                                                  TRAP    C$PNTF
031666  062706  000012                                                         ADD     #12,SP
031672  016337  000010  002260          MOV D.BB(R3),TEMP
031700                                  GMANID T4BB,TEMP,D,-1,0,16.,YES ;NUMBER OF BAD BLOCKS
031700  104443                                                                  TRAP    C$GMAN
031702  000406                                                                  BR      10002$
031704  002260                                                                  .WORD   TEMP
031706  000052                                                                  .WORD   T$CODE
031710  003573                                                                  .WORD   T4BB
031712  177777                                                                  .WORD   -1
031714  000000                                                                  .WORD   T$LOLIM
031716  000020                                                                  .WORD   T$HILIM
031720                                                                  10002$:
031720  013763  002260  000010          MOV TEMP,D.BB(R3)
031726  001424                          BEQ T4Q02
                                ;
031730  010304                          MOV R3,R4               ;GET POINTER TO STORAGE
032732  062704  000012                  ADD #D.BB01,R4           ;  FOR BAD BLOCKS
032736  013701  002260                  MOV TEMP,R1             ;GET COUNT OF BLOCKS TO INPUT
                                ;
031742  004737  020126          T4Q01:  CALL BLD28              ;BUILD DEFAULT ANSWER
031746                                  GMANID T4BBI,TEMP,A,-1,0,9.,YES ;BAD BLOCK
031746  104443                                                                  TRAP    C$GMAN
031750  000406                                                                  BR      10003$
031752  002260                                                                  .WORD   TEMP
031754  000152                                                                  .WORD   T$CODE
031756  003620                                                                  .WORD   T4BBI
031760  177777                                                                  .WORD   -1
031762  000000                                                                  .WORD   T$LOLIM
031764  000011                                                                  .WORD   T$HILIM
031766                                                                  10003$:
031766  004737  020230                  CALL CNV28              ;CONVERT TO BINARY
031772  103763                          BCS T4Q01               ;REPEAT UNTIL RIGHT
031774  005301                          DEC R1                  ;DECREMENT COUNT
031776  001361                          BNE T4Q01               ;GET ALL NUMBERS
032000  016337  000004  002260  T4Q02:  MOV D.PRM(R3),TEMP      ;GET PARAMETER BITS
032006                                  GMANIL T4RO,TEMP,D.RO,YES       ;READ ONLY
```

```
        032006  104443                                                          TRAP    C$GMAN
        032010  000404                                                          BR      10004$
        032012  002260                                                          .WORD   TEMP
        032014  000130                                                          .WORD   T$CODE
        032016  003632                                                          .WORD   T4RO
        032020  004000                                                          .WORD   D.RO
        032022                                                          10004$:
    30  032022  032737  004000  002260        BIT #D.RO,TEMP
    31  032030  001056                        BNE T4Q06
    32  032032                        T4Q03:  GMANIL T4WO,TEMP,D.WO,YES       ;WRITE ONLY
        032032  104443                                                          TRAP    C$GMAN
        032034  000404                                                          BR      10005$
        032036  002260                                                          .WORD   TEMP
        032040  000130                                                          .WORD   T$CODE
        032042  003644                                                          .WORD   T4WO
        032044  002000                                                          .WORD   D.WO
        032046                                                          10005$:
    33  032046                        GMANIL T4WCA,TEMP,D.WCA,YES     ;CHECK ALL WRITES
        032046  104443                                                          TRAP    C$GMAN
        032050  000404                                                          BR      10006$
        032052  002260                                                          .WORD   TEMP
        032054  000130                                                          .WORD   T$CODE
        032056  003657                                                          .WORD   T4WCA
        032060  000004                                                          .WORD   D.WCA
        032062                                                          10006$:
    34  032062  032737  000004  002260        BIT #D.WCA,TEMP                 ;CHECK ANSWER
    35  032070  001007                        BNE T4Q04                       ;BRANCH IF YES
    36  032072                                GMANIL T4WCR,TEMP,D.WC,YES      ;RANDOMLY CHECK WRITES
        032072  104443                                                          TRAP    C$GMAN
        032074  000404                                                          BR      10007$
        032076  002260                                                          .WORD   TEMP
        032100  000130                                                          .WORD   T$CODE
        032102  003713                                                          .WORD   T4WCR
        032104  000010                                                          .WORD   D.WC
        032106                                                          10007$:
    37  032106  000403                        BR T4Q05
    38  032110  052737  000010  002260  T4Q04: BIS #D.WC,TEMP                  ;BOTH BITS GET SET
    39  032116  013763  002260  000004  T4Q05: MOV TEMP,D.PRM(R3)              ;PUT PARAM BITS BACK
    40  032124  016337  000006  002260        MOV D.PAT(R3),TEMP
    41  032132                                GMANID T4DP,TEMP,D,-1,0,16.,YES ;DATA PATTERN
        032132  104443                                                          TRAP    C$GMAN
        032134  000406                                                          BR      10010$
        032136  002260                                                          .WORD   TEMP
        032140  000052                                                          .WORD   T$CODE
        032142  003754                                                          .WORD   T4DP
        032144  177777                                                          .WORD   -1
        032146  000000                                                          .WORD   T$LOLIM
        032150  000020                                                          .WORD   T$HILIM
        032152                                                          10010$:
    42  032152  013763  002260  000006        MOV TEMP,D.PAT(R3)
    43  032160  016337  000004  002260        MOV D.PRM(R3),TEMP              ;GET PARAM BITS AGAIN
    44  032166  032737  004000  002260  T4Q06: BIT #D.RO,TEMP                  ;BYPASS NEXT 3 IF ONLY WRITING
    45  032174  001010                        BNE T4Q07
    46  032176  032737  002000  002260        BIT #D.WO,TEMP
    47  032204  001404                        BEQ T4Q07
    48  032206  032737  000010  002260        BIT #D.WC,TEMP
    49  032214  001432                        BEQ T4Q09
```

```
     50 032216                          T4Q07:  GMANIL T4ECC,TEMP,D.ECC,YES     ;ENABLE ECC
        032216  104443                                                           TRAP    C$GMAN
        032220  000404                                                           BR      10011$
        032222  002260                                                           .WORD   TEMP
        032224  000130                                                           .WORD   T$CODE
        032226  004022                                                           .WORD   T4ECC
        032230  010000                                                           .WORD   D.ECC
        032232                                                              10011$:
     51 032232                                  GMANIL T4DCA,TEMP,D.DCA,YES     ;COMPARE ALL DATA
        032232  104443                                                           TRAP    C$GMAN
        032234  000404                                                           BR      10012$
        032236  002260                                                           .WORD   TEMP
        032240  000130                                                           .WORD   T$CODE
        032242  004055                                                           .WORD   T4DCA
        032244  000001                                                           .WORD   D.DCA
        032246                                                              10012$:
     52 032246  032737  000001  002260          BIT #D.DCA,TEMP                 ;CHECK ANSWER
     53 032254  001007                          BNE T4Q08                       ;BRANCH IF YES
     54 032256                                  GMANIL T4DCR,TEMP,D.DC,YES      ;RANDOMLY CHECK WRITES
        032256  104443                                                           TRAP    C$GMAN
        032260  000404                                                           BR      10013$
        032262  002260                                                           .WORD   TEMP
        032264  000130                                                           .WORD   T$CODE
        032266  004103                                                           .WORD   T4DCR
        032270  000002                                                           .WORD   D.DC
        032272                                                              10013$:
     55 032272  000403                          BR T4Q09
     56 032274  052737  000002  002260  T4Q08:  BIS #D.DC,TEMP                  ;BOTH BITS GET SET
     57 032302                          T4Q09:  GMANIL T4RET,TEMP,D.RET,YES     ;ENABLE RETRIES
        032302  104443                                                           TRAP    C$GMAN
        032304  000404                                                           BR      10014$
        032306  002260                                                           .WORD   TEMP
        032310  000130                                                           .WORD   T$CODE
        032312  004136                                                           .WORD   T4RET
        032314  001000                                                           .WORD   D.RET
        032316                                                              10014$:
     58 032316  005137  002260                  COM TEMP
     59 032322                                  GMANIL T4SEK,TEMP,D.SEQ,YES     ;ENABLE SEEKS
        032322  104443                                                           TRAP    C$GMAN
        032324  000404                                                           BR      10015$
        032326  002260                                                           .WORD   TEMP
        032330  000130                                                           .WORD   T$CODE
        032332  004155                                                           .WORD   T4SEK
        032334  000100                                                           .WORD   D.SEQ
        032336                                                              10015$:
     60 032336  005137  002260                  COM TEMP                        ;COMPLIMENTED
     61 032342  013763  002260  000004          MOV TEMP,D.PRM(R3)
     62
     63 032350  005037  002260                  CLR TEMP                        ;DETERMINE DEFAULT SELECTION
     64 032354  032763  000040  000004          BIT #D.BE,D.PRM(R3)             ;IF D.BE SET - LOAD 1
     65 032362  001403                          BEQ T4Q10                       ;IF D.CYL CLEAR - LOAD 0
     66 032364  005237  002260                  INC TEMP                        ;IF D.BEC CONTAINS 0 - LOAD 4
     67 032370  000422                          BR T4Q11                        ;IF D.TR SET - LOAD 2
     68 032372  032763  000400  000004  T4Q10:  BIT #D.CYL,D.PRM(R3)            ;LOAD 3
     69 032400  001416                          BEQ T4Q11
     70 032402  012737  000004  002260          MOV #4,TEMP
     71 032410  005763  000112                  TST D.BEC(R3)
```

```
     72 032414  001410                          BEQ T4Q11
     73 032416  005337  002260                  DEC TEMP
     74 032422  032763  000020  000004          BIT #D.TR,D.PRM(R3)
     75 032430  001402                          BEQ T4Q11
     76 032432  005337  002260                  DEC TEMP
     77 032436                          T4Q11:  PRINTF #T4OPT1
        032436  012746  007432                                          MOV     #T4OPT1,-(SP)
        032442  012746  000001                                          MOV     #1,-(SP)
        032446  010600                                                  MOV     SP,R0
        032450  104417                                                  TRAP    C$PNTF
        032452  062706  000004                                          ADD     #4,SP
     78 032456                                  PRINTF #T4OPT2
        032456  012746  007460                                          MOV     #T4OPT2,-(SP)
        032462  012746  000001                                          MOV     #1,-(SP)
        032466  010600                                                  MOV     SP,R0
        032470  104417                                                  TRAP    C$PNTF
        032472  062706  000004                                          ADD     #4,SP
     79 032476                                  PRINTF #T4OPT3
        032476  012746  007525                                          MOV     #T4OPT3,-(SP)
        032502  012746  000001                                          MOV     #1,-(SP)
        032506  010600                                                  MOV     SP,R0
        032510  104417                                                  TRAP    C$PNTF
        032512  062706  000004                                          ADD     #4,SP
     80 032516                                  PRINTF #T4OPT4
        032516  012746  007577                                          MOV     #T4OPT4,-(SP)
        032522  012746  000001                                          MOV     #1,-(SP)
        032526  010600                                                  MOV     SP,R0
        032530  104417                                                  TRAP    C$PNTF
        032532  062706  000004                                          ADD     #4,SP
     81 032536                                  PRINTF #T4OPT5
        032536  012746  007657                                          MOV     #T4OPT5,-(SP)
        032542  012746  000001                                          MOV     #1,-(SP)
        032546  010600                                                  MOV     SP,R0
        032550  104417                                                  TRAP    C$PNTF
        032552  062706  000004                                          ADD     #4,SP
     82 032556                                  PRINTF #T4OPT6
        032556  012746  007737                                          MOV     #T4OPT6,-(SP)
        032562  012746  000001                                          MOV     #1,-(SP)
        032566  010600                                                  MOV     SP,R0
        032570  104417                                                  TRAP    C$PNTF
        032572  062706  000004                                          ADD     #4,SP
     83 032576                                  GMANID T4OPT7,TEMP,D,-1,0,4,YES ;WHICH SELECTION LIMITS
        032576  104443                                                  TRAP    C$GMAN
        032600  000406                                                  BR      10016$
        032602  002260                                                  .WORD   TEMP
        032604  000052                                                  .WORD   T$CODE
        032606  004176                                                  .WORD   T4OPT7
        032610  177777                                                  .WORD   -1
        032612  000000                                                  .WORD   T$LOLIM
        032614  000004                                                  .WORD   T$HILIM
        032616                                                                  10016$:
     84 032616  005337  002260                  DEC TEMP                ;SET UP D.PRM FROM ANSWER
     85 032622  002004                          BGE T4Q12               ;IF 0 - CLEAR D.BE AND D.CYL
     86 032624  042763  000440  000004          BIC #D.BE+D.CYL,D.PRM(R3)
     87 032632  000467                          BR T4Q19
     88 032634  005337  002260          T4Q12:  DEC TEMP                ;IF 1
     89 032640  002013                          BGE T4Q13               ;  IF D.BE NOT SET
```

```
     90 032642  032763  000040  000004          BIT #D.BE,D.PRM(R3)             ;    SET D.BE
     91 032650  001060                          BNE T4Q19                       ;    CLEAR D.CYL
     92 032652  052763  000040  000004          BIS #D.BE,D.PRM(R3)             ;    LOAD 1 IN D.BEC
     93 032660  042763  000400  000004          BIC #D.CYL,D.PRM(R3)            ;    CLEAR BLOCK STORAGE
     94 032666  000436                          BR T4Q16
     95 032670  042763  000040  000004  T4Q13:  BIC #D.BE,D.PRM(R3)            ;IF 2, 3 OR 4
     96                                                                         ;  CLEAR D.BE
     97 032676  022737  000002  002260          CMP #2,TEMP                     ;IF 4
     98 032704  001006                          BNE T4Q14                       ;  SET D.CYL
     99 032706  052763  000400  000004          BIS #D.CYL,D.PRM(R3)            ;  CLEAR D.BEC
    100 032714  005063  000112                  CLR D.BEC(R3)
    101 032720  000434                          BR T4Q19
    102 032722                          T4Q14:  PUSH D.PRM(R3)                  ;IF 2 OR 3
        032722  016346  000004                          MOV D.PRM(R3),-(SP)
    103 032726  052763  000420  000004          BIS #D.CYL+D.TR,D.PRM(R3)       ;  SAVE D.PRM BITS
    104 032734  005337  002260                  DEC TEMP                        ;  SET D.CYL AND D.TR
    105 032740  100403                          BMI T4Q15                       ;  IF 3
    106 032742  042763  000020  000004          BIC #D.TR,D.PRM(R3)             ;    CLEAR D.TR
    107 032750  022663  000004          T4Q15:  CMP (SP)+,D.PRM(R3)             ;  IF D.CYL OR D.TR CHANGED OR D.BEC = 0
    108 032754  001003                          BNE T4Q16
    109 032756  005763  000112                  TST D.BEC(R3)                   ;    LOAD 1 IN D.BEC
    110 032762  001013                          BNE T4Q19                       ;    CLEAR BLOCK STORAGE
    111 032764  012763  000001  000112  T4Q16:  MOV #1,D.BEC(R3)
    112 032772  010304                  T4Q17:  MOV R3,R4
    113 032774  062704  000114                  ADD #D.BGN1,R4
    114 033000  012701  000020                  MOV #16.,R1
    115 033004  005024                  T4Q18:  CLR (R4)+
    116 033006  005301                          DEC R1
    117 033010  001375                          BNE T4Q18
    118 033012  032763  000040  000004  T4Q19:  BIT #D.BE,D.PRM(R3)            ;NOW ASK THE QUESTIONS TO ALLOW THE
    119 033020  001460                          BEQ T4Q22                       ;  NUMBERS TO CHANGE
    120 033022  016337  000112  002260          MOV D.BEC(R3),TEMP
    121 033030                                  GMANID T4BE,TEMP,D,-1,1,4,YES   ;NUMBER OF B/E SETS
        033030  104443                                                                  TRAP    C$GMAN
        033032  000406                                                                  BR      10017$
        033034  002260                                                                  .WORD   TEMP
        033036  000052                                                                  .WORD   T$CODE
        033040  004201                                                                  .WORD   T4BE
        033042  177777                                                                  .WORD   -1
        033044  000001                                                                  .WORD   T$LOLIM
        033046  000004                                                                  .WORD   T$HILIM
        033050                                                                  10017$:
    122 033050  013763  002260  000112          MOV TEMP,D.BEC(R3)
    123 033056  013701  002260                  MOV TEMP,R1                     ;GET COUNT OF SETS
    124 033062  010304                          MOV R3,R4                       ;GET POINTER TO STORAGE AREA
    125 033064  062704  000114                  ADD #D.BGN1,R4
    126 033070  004737  020126          T4Q20:  CALL BLD28
    127 033074                                  GMANID T4BEG,TEMP,A,-1,0,9.,YES ;BEGIN BLOCK
        033074  104443                                                                  TRAP    C$GMAN
        033076  000406                                                                  BR      10020$
        033100  002260                                                                  .WORD   TEMP
        033102  000152                                                                  .WORD   T$CODE
        033104  004232                                                                  .WORD   T4BEG
        033106  177777                                                                  .WORD   -1
        033110  000000                                                                  .WORD   T$IOLIM
        033112  000011                                                                  .WORD   T$HILIM
        033114                                                                  10020$:
```

```
   128 033114  004737  020230                  CALL CNV28
   129 033120  103763                          BCS T4Q20
   130 033122  004737  020126          T4Q21:  CALL BLD28
   131 033126                                  GMANID T4END,TEMP,A,-1,0,9.,YES ;END BLOCK
       033126  104443                                                          TRAP    C$GMAN
       033130  000406                                                          BR      10021$
       033132  002260                                                          .WORD   TEMP
       033134  000152                                                          .WORD   T$CODE
       033136  004246                                                          .WORD   T4END
       033140  177777                                                          .WORD   -1
       033142  000000                                                          .WORD   T$LOLIM
       033144  000011                                                          .WORD   T$HILIM
       033146                                                          10021$:
   132 033146  004737  020230                  CALL CNV28
   133 033152  103763                          BCS T4Q21
   134 033154  005301                          DEC R1
   135 033156  001344                          BNE T4Q20
   136 033160  000577                          BR T4Q30
   137 033162  032763  000400  000004  T4Q22:  BIT #D.CYL,D.PRM(R3)          ;IF D.CYL CLEAR - ALL DONE
   138 033170  001573                          BEQ T4Q30
   139 033172  005763  000112                  TST D.BEC(R3)                ;IF D.BEC CLEAR - GO RIGHT TO B/E CYLS
   140 033176  001526                          BEQ T4Q27
   141 033200  010304                          MOV R3,R4
   142 033202  062704  000112                  ADD #D.BEC,R4
   143 033206  032763  000020  000004          BIT #D.TR,D.PRM(R3)          ;LOOK AT D.TR.TO DETERMINE QUESTION
   144 033214  001434                          BEQ T4Q24
   145 033216  011437  002260                  MOV (R4),TEMP
   146 033222                                  GMANID T4TRC,TEMP,D,-1,1,7,YES  ;NUMBER OF TRACKS
       033222  104443                                                          TRAP    C$GMAN
       033224  000406                                                          BR      10022$
       033226  002260                                                          .WORD   TEMP
       033230  000052                                                          .WORD   T$CODE
       033232  004260                                                          .WORD   T4TRC
       033234  177777                                                          .WORD   -1
       033236  000001                                                          .WORD   T$LOLIM
       033240  000007                                                          .WORD   T$HILIM
       033242                                                          10022$:
   147 033242  013714  002260                  MOV TEMP,(R4)
   148 033246  012401                          MOV (R4)+,R1                 ;GET COUNT OF TRACKS
   149 033250  011437  002260          T4Q23:  MOV (R4),TEMP
   150 033254                                  GMANID T4TRAK,TEMP,D,-1,0,255.,YES      ;TRACK
       033254  104443                                                          TRAP    C$GMAN
       033256  000406                                                          BR      10023$
       033260  002260                                                          .WORD   TEMP
       033262  000052                                                          .WORD   T$CODE
       033264  004311                                                          .WORD   T4TRAK
       033266  177777                                                          .WORD   -1
       033270  000000                                                          .WORD   T$LOLIM
       033272  000377                                                          .WORD   T$HILIM
       033274                                                          10023$:
   151 033274  013724  002260                  MOV TEMP,(R4)+
   152 033300  005301                          DEC R1
   153 033302  001362                          BNE T4Q23
   154 033304  000433                          BR T4Q26
   155 033306  011437  002260          T4Q24:  MOV (R4),TEMP
   156 033312                                  GMANID T4GRC,TEMP,D,-1,1,7,YES  ;NUMBER OF GROUPS
       033312  104443                                                          TRAP    C$GMAN
```

```
        033314  000406                                                          BR      10024$
        033316  002260                                                          .WORD   TEMP
        033320  000052                                                          .WORD   T$CODE
        033322  004317                                                          .WORD   T4GRC
        033324  177777                                                          .WORD   -1
        033326  000001                                                          .WORD   T$LOLIM
        033330  000007                                                          .WORD   T$HILIM
        033332                                                          10024$:
    157 033332  013714  002260          MOV TEMP,(R4)
    158 033336  012401                  MOV (R4)+,R1                    ;GET COUNT OF GROUPS
    159 033340  011437  002260  T4Q25:  MOV (R4),TEMP
    160 033344                          GMANID T4GRP,TEMP,D,-1,0,255.,YES       ;GROUP
        033344  104443                                                          TRAP    C$GMAN
        033346  000406                                                          BR      10025$
        033350  002260                                                          .WORD   TEMP
        033352  000052                                                          .WORD   T$CODE
        033354  004350                                                          .WORD   T4GRP
        033356  177777                                                          .WORD   -1
        033360  000000                                                          .WORD   T$LOLIM
        033362  000377                                                          .WORD   T$HILIM
        033364                                                          10025$:
    161 033364  013724  002260          MOV TEMP,(R4)+
    162 033370  005301                  DEC R1
    163 033372  001362                  BNE T4Q25
    164 033374  016337  000162  002260  T4Q26:  MOV D.ECYL+2(R3),TEMP
    165 033402  005137  002260          COM TEMP
    166 033406                          GMANIL T4CYL,TEMP,BIT15,YES     ;WISH TO LIMIT CYLINDERS
        033406  104443                                                          TRAP    C$GMAN
        033410  000404                                                          BR      10026$
        033412  002260                                                          .WORD   TEMP
        033414  000130                                                          .WORD   T$CODE
        033416  004356                                                          .WORD   T4CYL
        033420  100000                                                          .WORD   BIT15
        033422                                                          10026$:
    167 033422  005737  002260          TST TEMP
    168 033426  100412                  BMI T4Q27
    169 033430  005063  000154          CLR D.BCYL(R3)
    170 033434  005063  000156          CLR D.BCYL+2(R3)
    171 033440  005063  000160          CLR D.ECYL(R3)
    172 033444  012763  177777  000162  .       MOV #-1,D.ECYL+2(R3)
    173 033452  000442                  BR T4Q30
    174 033454  005763  000162  T4Q27:  TST D.ECYL+2(R3)
    175 033460  002002                  BGE T4Q27A
    176 033462  005063  000162          CLR D.ECYL+2(R3)
    177 033466  010304          T4Q27A: MOV R3,R4
    178 033470  062704  000154          ADD #D.BCYL,R4
    179 033474  004737  020126  T4Q28:  CALL BLD28
    180 033500                          GMANID T4CYLB,TEMP,A,-1,0,9.,YES        ;STARTING CYLINDER
        033500  104443                                                          TRAP    C$GMAN
        033502  000406                                                          BR      10027$
        033504  002260                                                          .WORD   TEMP
        033506  000152                                                          .WORD   T$CODE
        033510  004430                                                          .WORD   T4CYLB
        033512  177777                                                          .WORD   -1
        033514  000000                                                          .WORD   T$ILOLIM
        033516  000011                                                          .WORD   T$HILIM
        033520                                                          10027$:
```

```
   181 033520  004737  020230                  CALL CNV28
   182 033524  103763                          BCS T4Q28
   183 033526  004737  020126          T4Q29:  CALL BLD28
   184 033532                                  GMANID T4CYLE,TEMP,A,-1,0,9.,YES        ;ENDING CYLINDER
       033532  104443                                                                      RAP     C$GMAN
       033534  000406                                                                     BR      10030$
       033536  002260                                                                     .WORD   TEMP
       033540  000152                                                                     .WORD   T$CODE
       033542  004452                                                                     .WORD   T4CYLE
       033544  177777                                                                     .WORD   -1
       033546  000000                                                                     .WORD   T$LOLIM
       033550  000011                                                                     .WORD   T$HILIM
       033552                                                                          10030$:
   185 033552  004737  020230                  CALL CNV28
   186 033556  103763                          BCS T4Q29
   187 033560                          T4Q30:  POP <R4,R2>
       033560  012604                                  MOV (SP)+,R4
       033562  012602                                  MOV (SP)+,R2
   188 033564  000207                          RETURN
   189 033566                                  ENDTST
       033566                                                                          L10043:
       033566  104401                                                                      TRAP    C$ETST
   190 033570                                  ENDMOD
   191                                 .SBTTL
```

```
                                ;.TITLE PARAMETER CODING   (6)
                                .SBTTL  HARDWARE PARAMETER CODING SECTION
033570                                  BGNMOD
                                ;++
                                ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ; WITH THE OPERATOR.
                                ;--
033570                                  BGNHRD
033570  000032                                                          .WORD L10044-L$HARD/2
033572                                                          L$HARD::
                                ;FORMAT OF HARDWARE P-TABLE IS AS FOLLOWS:
000000                          HO.UBA  ==      0.                      ; UNIBUS ADDRESS
000002                          HO.VEC  ==      HO.UBA+2                ; UDA VECTOR
000004                          HO.BRL  ==      HO.VEC+2                ; BR LEVEL
000006                          HO.BST  ==      HO.BRL+2                ; BURST RATE
000010                          HO.LDR  ==      HO.BST+2                ; DRIVE NUMBER
000012                          HO.PRM  ==      HO.LDR+2                ; PROGRAM PARAMETERS
020000                                  HM.CYL  ==      BIT13           ; TEST CUSTOMER DATA AREA
```

```
HARDWARE PARAMETER CODING SECTION

      1 033572                          GPRMA   H.UBA,HO.UBA,0,160000,177774,YES        ;BUS ADDRESS
        033572  000031                                                                          .WORD   T$CODE
        033574  033656                                                                          .WORD   H.UBA
        033576  160000                                                                          .WORD   T$LOLIM
        033600  177774                                                                          .WORD   T$HILIM
      2 033602                          GPRMA   H.VEC,HO.VEC,0,4,774,YES         ; VECTOR
        033602  001031                                                                          .WORD   T$CODE
        033604  033704                                                                          .WORD   H.VEC
        033606  000004                                                                          .WORD   T$LOLIM
        033610  000774                                                                          .WORD   T$HILIM
      3 033612                          GPRMD   H.BRL,HO.BRL,D,-1,4.,7.,YES      ; BR LEVEL
        033612  002052                                                                          .WORD   T$CODE
        033614  033713                                                                          .WORD   H.BRL
        033616  177777                                                                          .WORD   -1
        033620  000004                                                                          .WORD   T$LOLIM
        033622  000007                                                                          .WORD   T$HILIM
      4 033624                          GPRMD   H.BST,HO.BST,D,-1,0.,63.,YES     ; BURST RATE
        033624  003052                                                                          .WORD   T$CODE
        033626  033724                                                                          .WORD   H.BST
        033630  177777                                                                          .WORD   -1
        033632  000000                                                                          .WORD   T$LOLIM
        033634  000077                                                                          .WORD   T$HILIM
      5 033636                          GPRMD   H.LDR,HO.LDR,D,-1,0.,255.,YES    ; DRIVE SELECT NUMBER
        033636  004052                                                                          .WORD   T$CODE
        033640  033746                                                                          .WORD   H.LDR
        033642  177777                                                                          .WORD   -1
        033644  000000                                                                          .WORD   T$LOLIM
        033646  000377                                                                          .WORD   T$HILIM
      6 033650                          GPRML   H.CST,HO.PRM,HM.CYL,YES ; USE CUSTOMER DATA AREA
        033650  005130                                                                          .WORD   T$CODE
        033652  033763                                                                          .WORD   H.CST
        033654  020000                                                                          .WORD   HM.CYL
      7 033656                          ENDHRD
                                                                                                .EVEN
        033656                                                                  L10044:
      8
     15 033656      125     116     111  H.UBA:  .ASCIZ  \UNIBUS ADDRESS OF UDA\
     16 033704      126     105     103  H.VEC:  .ASCIZ  \VECTOR\
     17 033713      102     122     040  H.BRL:  .ASCIZ  \BR LEVEL\
     18 033724      125     116     111  H.BST:  .ASCIZ  \UNIBUS BURST RATE\
     19 033746      104     122     111  H.LDR:  .ASCIZ  \DRIVE NUMBER\
     20 033763      105     130     105  H.CST:  .ASCIZ  \EXERCISE ON CUSTOMER DATA AREA IN TEST 4\
     21                                      .EVEN
```

```
                                .SBTTL  SOFTWARE PARAMETER CODING SECTION

                                ;++
                                ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ; WITH THE OPERATOR.
                                ;--

034034                                  BGNSFT
034034  000030                                                          .WORD L10045-L$SOFT/2
034036                                                          L$SOFT::

                                ;FORMAT OF SOFTWARE P-TABLE IS AS FOLLOWS:

        000000                  SO.EL=  0.                      ;ERROR LIMIT
        000002                  SO.XL=  2.                      ;DATA TRANSFER LIMIT (MEGABITS)
        000004                  SO.BIT= 4.                      ;SINGLE BIT ANSWERS
        000200                          SM.MAN==BIT07           ;  MANUAL INTERVENTION MODE
        000400                          SM.SSF==BIT08           ;  SUPPRESS SOFT ERRORS
        001000                          SM.LOG==BIT09           ;  ERROR LOG ENABLED
        040000                          SM.IW== BIT14           ;  INITIAL WRITE
```

```
      1 034036                          GPRML S.MAN,SO.BIT,SM.MAN,YES   ;MANUAL INTERVENTION MODE
        034036  002130                                                          .WORD   T$CODE
        034040  034116                                                          .WORD   S.MAN
        034042  000200                                                          .WORD   SM.MAN
      2 034044                          DISPLAY S.MES                   ;MESSAGE ON NEXT QUESTIONS
        034044  000003                                                          .WORD   T$CODE
        034046  034203                                                          .WORD   S.MES
      3 034050                          GPRMD S.EL,SO.EL,D,-1,1.,-1.,YES        ;ERROR LIMIT
        034050  000052                                                          .WORD   T$CODE
        034052  034266                                                          .WORD   S.EL
        034054  177777                                                          .WORD   -1
        034056  000001                                                          .WORD   T$LOLIM
        034060  177777                                                          .WORD   T$HILIM
      4 034062                          GPRMD S.XL,SO.XL,D,-1,0.,-1.,YES        ;TRANSFER LIMIT
        034062  001052                                                          .WORD   T$CODE
        034064  034302                                                          .WORD   S.XL
        034066  177777                                                          .WORD   -1
        034070  000000                                                          .WORD   T$LOLIM
        034072  177777                                                          .WORD   T$HILIM
      5 034074                          GPRML S.SSF,SO.BIT,SM.SSF,YES   ;SUPPRESS SOFT ERRORS
        034074  002130                                                          .WORD   T$CODE
        034076  034364                                                          .WORD   S.SSF
        034100  000400                                                          .WORD   SM.SSF
      6 034102                          GPRML S.IW,SO.BIT,SM.IW,YES     ;INITIAL WRITE
        034102  002130                                                          .WORD   T$CODE
        034104  034422                                                          .WORD   S.IW
        034106  040000                                                          .WORD   SM.IW
      7 034110                          GPRML S.LOG,SO.BIT,SM.LOG,YES   ;ERROR LOG
        034110  002130                                                          .WORD   T$CODE
        034112  034454                                                          .WORD   S.LOG
        034114  001000                                                          .WORD   SM.LOG
     11 034116                          ENDSFT
                                                                                .EVEN
        034116                                                          L10045:
     12
     13
     20 034116      105     116     124  S.MAN:   .ASCIZ\ENTER MANUAL INTERVENTION MODE FOR SPECIAL DIAGNOSIS\
     21 034203      122     105     115  S.MES:   .ASCIZ\REMAINING SOFTWARE QUESTIONS APPLY TO TEST 4 ONLY\
     22 034265      000                  .BYTE 0
     23 034266      105     122     122  S.EL:    .ASCIZ\ERROR LIMIT\
     24 034302      122     105     101  S.XL:    .ASCIZ\READ TRANSFER LIMIT IN MEGABYTES - 0 FOR NO LIMIT\
     25 034364      123     125     120  S.SSF:   .ASCIZ\SUPPRESS PRINTING SOFT ERRORS\
     26 034422      104     117     040  S.IW:    .ASCIZ\DO INITIAL WRITE ON START\
     27 034454      105     116     101  S.LOG:   .ASCIZ\ENABLE ERROR LOG\
     31                                  .EVEN
```

PATCH AREA

```
                                        .SBTTL  PATCH AREA
                                
034476                          $PATCH::
        000050                          .REPT 40.
                                        .WORD 0
                                        .ENDR
                                
034616                                  LASTAD
                                                                        .EVEN
034616  034642                                                          .WORD T$FREE
034620  000010                                                          .WORD T$SIZE
034622                          L$LAST::
034622                                  ENDMOD
                                .SBTTL
```

```
      13 034622                                  BGNSETUP        1
      14 034622                                  BGNPTAB
         034622  000000                                                                .WORD   0
         034624  000006                                                                .WORD   L10050-./2-1
         034626                                                                     L10046:
      15 034626  172150                          .WORD   172150          ; UNIBUS ADDRESS
      16 034630  000154                          .WORD   154             ; VECTOR ADDRESS
      17 034632  000005                          .WORD   5.              ; BR LEVEL
      18 034634  000000                          .WORD   0               ; UNIBUS BURST RATE
      19 034636  000000                          .WORD   0.              ; LOGICAL DRIVE NUMBER
      20 034640  000000                          .WORD   0               ; CUSTOMER DATA AREA
      21 034642                                  ENDPTAB
         034642                                                                     L10050:
      22 034642                                  ENDSETUP
      23
      24                                 .NLIST SYM
      25         000001                  .END
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  29184 WORDS  ( 114 PAGES)
DYNAMIC MEMORY AVAILABLE FOR  71 PAGES
,B:ZUDCA0/C=[20,0]SVC34R.MLB/P:1,B:ZJDCA0.DOC,B:ZUDCA0
```

CROSS REFERENCE TABLE (CREF V01-05)

```
$PATCH 125-3#
ADR       9-70#
ALOCM    29-16#   30-15    93-72   104-4    105-20   107-4
APRINT   76-14#
APRIZ    76-17    76-19#
ASSEMB    5-16     5-16
BADT     25-13#  101-122
BITO      9-70#   19-25
BIT00     9-70     9-70#
BIT01     9-70     9-70#
BIT02     9-70     9-70#
BIT03     9-70     9-70#
BIT04     9-70     9-70#
BIT05     9-70     9-70#
BIT06     9-70     9-70#
BIT07     9-70     9-70#  123-27
BIT08     9-70     9-70#  123-28
BIT09     9-70     9-70#  123-29
BIT1      9-70#   18-21    19-24    22-40
BIT10     9-70#   19-16
BIT11     9-70#   19-15
BIT12     9-70#   19-14
BIT13     9-70#   19-13   121-70
BIT14     9-70#   19-12   123-30
BIT15     9-70#   18-13    19-10    19-27    33-25    49-31    50-23    50-27    51-30    52-45    61-29    99-79    99-87   112-25
         120-166
BIT2      9-70#   18-22    19-23    22-41
BIT3      9-70#   18-23    19-22    22-42
BIT4      9-70#   18-25    19-21    22-43
BIT5      9-70#   19-20
BIT6      9-70#   19-19
BIT7      9-70#   19-27
BIT8      9-70#   19-18
BIT9      9-70#   19-17
BLD28    63-8#   120-22   120-126  120-130  120-179  120-183
BLDC0    72-22    72-24#
BLDC1    72-26#   72-28
BLDCMD   36-10    36-29    68-44    70-17    72-15#
BOE       9-70#
BRLEV   116-15*  116-65   116-84#
C$AU      5-16#
C$AUTO    5-16#  110-18
C$BRK     5-16#   34-43    75-28    86-13    88-27    93-24    97-10   101-79   101-88   115-11   116-33   119-113
C$BSEG    5-16#
C$BSUB    5-16#  112-28   112-41   112-44   112-47   112-51   112-57
C$CEFG    5-16#
C$CLCK    5-16#  102-13   102-15
C$CLEA    5-16#  111-30
C$CLOS    5-16#   92-12
C$CLP1    5-16#
C$CVEC    5-16#   77-36    86-27   112-34   116-72   116-77
C$DCLN    5-16#   28-8     32-21    94-2     94-4    101-123  108-35   109-3    109-7    109-12
C$DODU    5-16#
C$DRPT    5-16#   34-54   101-73   119-116
C$DU      5-16#
C$EDIT    5-16#    5-77
```

CROSS REFERENCE TABLE (CREF V01-05)

```
C$ERDF    5-16#  34-22  34-40  59-24  75-36  75-44  85-18  85-37  86-30  87-44  88-36  89-5   112-37 114-17
        114-27   115-19 116-41 116-61 116-67
C$ERHR    5-16#  35-11  35-18  35-25  36-6   71-3
C$ERRO    5-16#  67-27
C$ERSF    5-16#  28-7   32-20  94-1   94-3   101-122 109-2  109-6  109-11
C$ERSO    5-16#
C$ESCA    5-16#
C$ESEG    5-16#
C$ESUB    5-16#  112-40 112-43 112-46 112-50 112-56 112-66
C$ETST    5-16#  113-6  117-24 118-14 120-189
C$EXIT    5-16#  108-45 119-117
C$GETB    5-16#  93-26  93-31  93-41  93-76  93-83  93-89  93-99  93-102
C$GETW    5-16#
C$GMAN    5-16#  41-66  108-32 119-64 119-68 120-14 120-23 120-29 120-32 120-33 120-36 120-41 120-50 120-51
        120-54   120-57 120-59 120-83 120-121 120-127 120-131 120-146 120-150 120-156 120-160 120-166 120-180 120-184
C$GPHR    5-16#  101-114 103-10 106-5
C$GPLO    5-16#
C$GPRI    5-16#
C$INIT    5-16#  109-15
C$INLP    5-16#
C$MANI    5-16#  41-32  108-30 119-36
C$MEM     5-16#  102-3
C$MSG     5-16#  27-17  27-22  27-28  27-32  27-37  27-49  27-54  27-59  27-63  27-67  27-71  27-75  27-79
        27-83    67-48
C$OPEN    5-16#  93-21
C$PNTB    5-16#  27-15  27-16  27-20  27-21  27-25  27-26  27-27  27-31  27-35  27-40  27-52  27-53  27-57
        27-58    27-62  27-66  27-70  27-74  27-78  27-82  67-34  67-37  98-23  98-24
C$PNTF    5-16#  41-34  41-42  41-43  41-44  41-45  41-53  41-59  41-62  41-113 49-33  49-34  51-60  51-63
        52-44    55-18  56-56  64-39  64-43  65-4   65-13  65-30  102-18 108-12 108-21 116-16 116-69 119-12
      119-19     119-24 119-76 120-12 120-77 120-78 120-79 120-80 120-81 120-82
C$PNTS    5-16#  99-59  99-70  99-71  99-74  99-75  99-88  99-107 99-108
C$PNTX    5-16#  27-36  27-41  27-42  27-45  53-21  56-6   56-14  56-43  56-64  58-13  58-20  61-59  61-69
C$QIO     5-16#
C$RDBU    5-16#
C$REFG    5-16#  101-63 101-65 101-69 101-91
C$RESE    5-16#  5-16#  97-11  102-1
C$REVI    5-16#  5-77
C$RFLA    5-16#  49-28  51-27  52-41
C$RPT     5-16#  99-156
C$SEFG    5-16#
C$SPRI    5-16#  108-42 116-18 116-44 116-55 116-63
C$SVEC    5-16#  68-32  77-17  86-24  102-24 112-30 116-17
C$TPRI    5-16#
C.BST     18-17# 85-42  105-9
C.DR0     18-29# 50-17  59-15  99-81  101-100 107-9  108-15 119-45 119-83
C.DR1     18-30#
C.DR2     18-31#
C.DR3     18-32#
C.DR4     18-33#
C.DR5     18-34#
C.DR6     18-35#
C.DR7     18-36#
C.FLG     18-20# 33-22* 34-9   34-12  34-14  34-68* 35-6   35-30  36-24* 36-25  36-32* 36-35* 49-32* 51-35*
        53-16* 68-57* 68-58* 70-27* 73-27* 75-19  75-40*
C.JAD     18-19#
C.JSR     18-18# 68-31
C.REF     18-39# 35-23  73-16* 73-17
```

CROSS REFERENCE TABLE (CREF V01-05)

```
C.RING   18-28#  30-17*  34-8     72-16    73-15
C.SIZE   18-41#  33-29   34-61    99-114   101-107  101-119  103-22   104-3    106-13   108-24   113-1    117-20   119-56   119-95
C.TO     18-37#  34-38   36-40    75-14    75-34    88-22    88-33   115-7    115-17   116-29   116-39
C.TOH    18-38#  34-35   75-31    88-30   115-14   116-36
C.UADR   18-10#  67-34   86-22    99-110  103-19   114-6    116-6    116-16
C.UNIT   18-11#  33-23   34-11    67-14    99-77   101-98*  101-127* 112-23
C.VEC    18-14#  68-28  105-7    116-7    116-70   116-75
CALR1    56-71   57-4#
CALR2    56-72   58-4#
CF.576   14-39#
CF.ATN   14-34#
CF.MSC   14-35#
CF.OTH   14-36#
CF.SHD   14-38#
CF.THS   14-37#
CLOG     84-26   84-34   90-10#
CLOGLP   90-13#  90-15
CLOSEF   91-14   92-10#  93-107   108-44   111-18
CLRBFL   74-22#  74-24
CLRBUF   36-12   36-31   74-16#
CNV28    64-15# 120-24  120-128  120-132  120-181  120-185
CON.A    56-12#  56-115
CON.A1   56-13#  56-16   58-19
CON.A2   56-18   56-20#
CON.D    56-24#  56-115
CON.H    56-30#  56-115
CON.N    56-42#  56-115
CON.N1   56-43#  56-45
CON.O    56-36#  56-115
CON.QU   56-3#   56-7    56-115
CON.QX   56-5    56-8#
CON.R    56-50#  56-115
CON.R1   56-52   56-56#
CON.S    56-62#  56-115
CON.S1   56-64#  56-66
CT.AVL   18-13#  33-25   99-79    101-98   101-127  112-25
CT.BRL   18-16# 116-10
CT.CMD   18-22#  34-14   68-57    70-27    73-27
CT.MSG   18-23#  34-12   34-68    36-24    49-32    51-35    53-16    75-19    75-40    82-20
CT.REQ   18-25#  35-6    35-30    36-25    36-32    36-35    68-57
CT.RN    18-21#  34-9    34-68    68-58
CT.UNT   18-12#
CT.VEC   18-15#  68-29  116-9    116-71   116-76
CTABE    27-73# 109-2
CTABEM   26-59#  27-74
CTABER  105-11  109-2#
CTABS    22-29#  99-76   101-96   101-116  103-3*   103-4*   103-15   106-10   108-13   112-21   117-8    118-8    119-42   119-80
        119-106
CTRLRS   22-30# 103-5*  104-19*  118-9    119-107
D.BB     19-29#  19-30   45-25    120-13   120-15*
D.BB01   19-30#  19-31  120-19
D.BB02   19-31#  19-32
D.BB03   19-32#  19-33
D.BB04   19-33#  19-34
D.BB05   19-34#  19-35
D.BB06   19-35#  19-36
D.BB07   19-36#  19-37
```

CROSS REFERENCE TABLE (CREF V01-05)

```
D.BB08  19-37#   19-38
D.BB09  19-38#   19-39
D.BB10  19-39#   19-40
D.BB11  19-40#   19-41
D.BB12  19-41#   19-42
D.BB13  19-42#   19-43
D.BB14  19-43#   19-44
D.BB15  19-44#   19-45    46-18
D.BB16  19-45#   19-46
D.BCYL  19-55#   19-56    44-44    44-62   120-169* 120-170* 120-178
D.BE    19-20#   44-17    44-58    44-70   120-64   120-86   120-90   120-92   120-95   120-118
D.BEC   19-46#   19-47    44-24    44-40    44-50   120-71   120-100* 120-109  120-111* 120-120  120-122* 120-139  120-142
D.BGN1  19-47#   19-48   120-113  120-125
D.BGN2  19-49#   19-50
D.BGN3  19-51#   19-52
D.BGN4  19-53#   19-54
D.CYL   19-18#   44-35   120-68   120-86   120-93   120-99   120-103  120-137
D.DC    19-24#   19-26   120-54   120-56
D.DCA   19-25#  120-51   120-52
D.DCY   19-13#  108-19   119-87
D.DRV   19-7#    19-8     99-109
D.ECC   19-14#   19-26   120-50
D.ECCC  19-62#   19-63    44-81*   47-18*   99-108
D.ECYL  19-56#   19-57   107-31*  120-164  120-171* 120-172* 120-174  120-176*
D.END1  19-48#   19-49
D.END2  19-50#   19-51
D.END3  19-52#   19-53
D.END4  19-54#   19-55
D.HERR  19-59#   19-60    44-78*   52-38*   52-39    99-108
D.IW    19-12#   19-27    44-14
D.PAT   19-28#   19-29    44-16   119-51   120-40   120-42*
D.PRM   19-11#   19-28    44-8     44-35   108-19   119-86   120-28   120-39*  120-43   120-61*  120-64   120-68   120-74   120-86*
        120-90  120-92*  120-93*  120-95*  120-99*  120-102  120-103* 120-106* 120-107  120-118  120-137  120-143
D.RET   19-17#   19-26   120-57
D.RO    19-15#  120-29   120-30   120-44
D.SEEK  19-61#   19-62    44-80*   48-15*   99-107
D.SEQ   19-19#  120-59
D.SERN  19-63#   19-65    44-5*    44-6*    44-7*    99-90    99-91    99-92
D.SERR  19-60#   19-61    44-79*   47-17*   99-108
D.SIZE  19-65#  107-3     107-27   107-31*
D.TR    19-21#  120-74   120-103  120-106  120-143
D.UNIT  19-8#    19-11    41-42    49-33    50-21    53-21    59-29    99-85    99-88    101-104* 101-128* 108-21   119-48   120-12
D.WC    19-22#   19-26   120-36   120-38   120-48
D.WCA   19-23#  120-33   120-34
D.WO    19-16#  120-32   120-46
D.XFRR  19-58#   19-59    44-77*   49-22*   49-26    99-107
D.XFRW  19-57#   19-58    44-76*   49-23*   99-107
D.ZERO  19-27#   44-9
DDEF    19-26#  107-26   119-88
DFPTBL   7-10#
DIAG    11-33#   72-23
DIAGMC   5-16     5-16
DIV10   99-99    99-120#
DIVIDE  61-39    61-52    62-16#   63-13    98-18    98-20    98-22    99-65    99-67    99-69
DLL     23-27#   40-37*   40-52*   40-53*   50-28   108-43*
DLLADR  23-32#   40-47*   40-48*   40-56    40-57
DLLDR   23-29#   40-38*   50-30    50-31
```

CROSS REFERENCE TABLE (CREF V01-05)

```
DLLNAM  23-34#   40-41*   40-42*   50-31    93-47    93-49
DLLR    23-31#   40-40*
DLLSIZ  23-33#   40-49*   40-54    40-55*   40-56*
DLLV    23-30#   40-39*
DMEND   22-33#   31-20    32-16*
DMENDS  22-34#   31-21    32-17*
DMFRST  20-37#   93-64    93-74
DMMAIN  20-36#   69-5     69-7     69-30
DMNF    25-46#   32-20
DMOVRL  20-35#
DMPROG  22-32#   32-13*   53-25    53-26    57-7     57-8     67-42    67-43    68-42    68-45    68-47    68-48    69-4     69-6
DMTRLN  20-34#
DONE    36-60    54-13#
DSPSIZ  36-4     36-62#
DT.AVL  19-10#   50-23    99-87    101-104  101-128  119-48
DT.UNT  19-9#
DUP     11-34#   72-20
E$END    5-16#
E$LOAD   5-16#    5-77
EF.BBR  14-27#
EF.BBU  14-28#
EF.CON   9-70#   101-63
EF.LOG  14-29#
EF.NEW   9-70#
EF.PWR   9-70#   101-69
EF.RES   9-70#   101-66
EF.SEX  14-30#
EF.STA   9-70#   101-91
ENG      5-47#    5-74     8-24     22-50    34-28    68-15    68-49    75-16    75-18    75-23    123-31   124-8    124-28
ERRBLK  22-21#
ERRC    55-13    55-20    56-103#
ERRCHR  22-23#   41-52*   41-53    41-58*   41-59    56-3*    56-4     56-6     56-13*   56-14    56-63*   56-64    61-58*   61-59
        61-68*   61-69    65-3*    65-4     65-12*   65-13    65-29*   65-30
ERRD    55-22    56-115#
ERRLIM  26-40#   52-44
ERRMB   26-35#   67-34
ERRMBD  26-36#   67-37
ERRMC   36-58    52-28#
ERRME1  26-38#   55-18    56-56
ERRMES  36-57    51-25#   52-29
ERRMRT  26-37#   98-23
ERRMSG  22-21#
ERRMSL  51-39    51-43#
ERRMSX  51-34    51-41    51-61    51-64#
ERRNBR  22-21#
ERRNL   26-21#   41-62    56-43    58-13    58-20    98-24    99-71
ERRONE  26-20#   41-53    41-59    56-6     56-14    56-64    61-59    61-69    65-4     65-13    65-30
ERRRSZ  56-51    56-73#
ERRRTB  56-54    56-71#   56-73
ERRRTN  67-26    67-31#
ERRTYP  22-21#   67-20
EVL      9-70#
F$AU     5-16#
F$AUTO   5-16#   110-10   110-18
F$BGN    5-16#    5-43     8-29     9-48     27-14    27-19    27-24    27-30    27-34    27-39    27-51    27-56    27-61    27-65
        27-69    27-73    27-77    27-81    67-31    81-10    82-18    95-5     96-1     98-28    99-38    99-45    100-8    101-8
       108-45   110-10   111-8    111-31   112-16   112-18   112-28   112-28   112-40   112-41   112-41   112-43   112-44   112-44
```

CROSS REFERENCE TABLE (CREF V01-05)

```
G$PRMD    5-16#   41-66   119-64  119-68  120-14  120-23  120-41  120-83  120-121 120-127 120-131 120-146 120-150 120-156
         120-160 120-180 120-184 122-3   122-4   122-5   124-3   124-4
G$PRML    5-16#  108-32  120-29  120-32  120-33  120-36  120-50  120-51  120-54  120-57  120-59  120-166 122-6   124-1
         124-5   124-6   124-7
G$RADA    5-16#   41-66   120-23  120-127 120-131 120-180 120-184
G$RADB    5-16#
G$RADD    5-16#  119-64  120-14  120-41  120-83  120-121 120-146 120-150 120-156 120-160 122-3   122-4   122-5   124-3
         124-4
G$RADL    5-16#  108-32  120-29  120-32  120-33  120-36  120-50  120-51  120-54  120-57  120-59  120-166 122-6   124-1
         124-5   124-6   124-7
G$RADO    5-16#  119-68  122-1   122-2
G$XFER    5-16#
G$YES     5-16#  119-64  119-68  120-14  120-23  120-29  120-32  120-33  120-36  120-41  120-50  120-51  120-54  120-57
         120-59  120-83  120-121 120-127 120-131 120-146 120-150 120-156 120-160 120-166 120-180 120-184 122-1   122-2
         122-3   122-4   122-5   122-6   124-1   124-3   124-4   124-5   124-6   124-7
GETCDN    60-16   60-18   60-28#
GETCNT    56-12   56-42   56-50   56-62   60-13#  61-15
GETCNX    60-15#  60-27
GETCXX    60-29   60-31#
GTDRV1    25-39#  59-24
GTDRVT    41-39   44-3    45-23   46-16   47-15   48-13   49-20   52-34   53-19   59-13#  67-18   101-125
H.BRL     122-3   122-17#
H.BST     122-4   122-18#
H.CST     122-6   122-20#
H.LDR     122-5   122-19#
H.UBA     122-1   122-15#
H.VEC     122-2   122-16#
HC.BF1    11-24#  11-25   36-11
HC.BF2    11-25#  11-27   36-3    36-30
HC.BSZ    11-13#  11-25   11-27   36-14   51-52   51-56   69-10   69-16   74-19   74-21   119-16
HC.CCT    11-19#  73-25*
HC.CEV    11-22#  11-23   72-18
HC.CMD    11-18#  11-19   11-20   30-21*
HC.CPK    11-23#  11-24   30-20   68-45*  68-46*  68-47*  68-48*  70-18*  70-19*  70-20*  70-21*  72-29*  73-17*  74-18*
         74-19*
HC.ESZ    11-11#  11-21   11-23   73-20
HC.INT    11-15#  11-16
HC.ISZ    11-9#   11-16   84-37
HC.MCT    11-17#  73-24*
HC.MEV    11-20#  11-21   73-18
HC.MPK    11-21#  11-22   30-18   30-20   35-9    35-16   35-23   68-55   70-25
HC.MSG    11-16#  11-17   11-18   30-19*  86-18
HC.PSZ    11-12#  11-22   11-24   72-19   72-25   73-20
HC.RSZ    11-10#  11-18   11-20
HC.SIZ    11-27#  30-14
HCOMM     30-14#  68-38
HELP       5-4#    5-11    5-33    5-57    5-80    6-10    7-12    8-12    9-3#    9-38    9-55    22-8    24-14   24-24
          26-3    26-13   99-5#   99-47   99-136  100-16  101-10  110-12  111-10  111-19  112-4#  121-4#  121-52  122-9
         123-14  124-14  125-7   126-1
HM.CYL    107-25  108-10  121-70#  122-6
HO.BRL    104-7   105-3   121-66#  121-67  122-3   122-3   122-3
HO.BST    104-12  105-9   121-67#  121-68  122-4   122-4   122-4
HO.LDR    101-124 107-13  107-20   121-68#  121-69  122-5   122-5   122-5
HO.PRM    107-22  121-69#  122-6   122-6   122-6
HO.UBA    103-20  121-64#  121-65  122-1   122-1   122-1
HO.VEC    104-10  105-6   121-65#  121-66  122-2   122-2   122-2
HOE        9-70#
```

CROSS REFERENCE TABLE (CREF V01-05)

```
I$AU      5-16#
I$AUTO    5-16#  110-10#  110-18#
I$CLN     5-16#  111-8#   111-30#
I$DU      5-16#
I$HRD   121-50#  122-7#   124-2
I$INIT    5-16#  101-8#   108-45   109-15#
I$MOD     5-16#    5-43     5-43#    8-29     8-29#    9-48     9-48#   98-28    98-28#   99-38    99-38#  111-31   111-31#  112-16
        112-16#  120-190  120-190# 121-39   121-39#  125-15   125-15#
I$MSG     5-16#   27-14#   27-17#   27-19#   27-22#   27-24#   27-28#   27-30#   27-32#   27-34#   27-37#   27-39#   27-49#   27-51#
         27-54#   27-56#   27-59#   27-61#   27-63#   27-65#   27-67#   27-69#   27-71#   27-73#   27-75#   27-77#   27-79#   27-81#
         27-83#   67-31#   67-48#
I$PROT    5-16#  100-8#
I$PTAB    5-16#  126-14   126-14#  126-21   126-21#
I$PWR     5-16#
I$RPT     5-16#   99-45#   99-156#
I$SEG     5-16#  112-18   112-28   112-41   112-44   112-47   112-51   112-57   117-3    118-3    119-3
I$SETU    5-16#  126-13   126-13#  126-14   126-22   126-22#
I$SFT   123-12#  124-2    124-11#
I$SRV     5-16#   81-10#   81-14#   82-18#   82-22#   95-5#    95-9#    96-1#    96-3#
I$SUB     5-16#  112-18   112-28   112-28#  112-40   112-40#  112-40#  112-41   112-41#  112-43   112-43#  112-43#  112-44   112-44#
        112-46   112-46#  112-46#  112-47   112-47#  112-50   112-50#  112-50#  112-51   112-51#  112-56   112-56#  112-56#  112-57
        112-57#  112-66   112-66#  112-66#  117-3    118-3    119-3
I$TST     5-16#  112-18   112-18#  112-28   112-41   112-44   112-47   112-51   112-57   113-6    113-6#   113-6#   117-3    117-3#
        117-24   117-24#  117-24#  118-3    118-3#   118-14   118-14#  118-14#  119-3    119-3#   119-117  120-189  120-189#  120-189#
IBE       9-70#
ICONT    22-40#  101-65   119-7
IDU       9-70#   49-29    51-28    52-42
IER       9-70#
IFLAGS   22-39#   44-10   101-65*  101-68*  102-6*   119-7    119-9*   119-32   119-101*  119-102*
INIT1   101-64   101-66#
INIT10  106-11#  106-14
INIT11  106-12   107-3#
INIT12  107-11#  107-17
INIT13  107-12   107-19#
INIT14  106-6    108-3#
INIT15  108-11   108-31   108-34   108-39#
INIT1A  101-67   101-69#
INIT2   101-70   101-81   101-85   101-91#
INIT3   101-92   102-1#
INIT3L  107-28#  107-30
INIT4   103-10#  105-17
INIT5   103-16#  103-23
INIT7   104-16#  104-18
INIT8   106-5#   108-5
INITC1  101-98#  101-109
INITC2  101-102# 101-106
INITC3  101-103  101-107#
INITC4  101-114# 101-131
INITC5  101-117# 101-121
INITC6  101-127#
INITC7  101-115  101-129#
INITCC  101-118  101-124#
INITE1  101-123# 101-126
INITKT   77-15#  101-72   102-2
INITMM   26-74#   27-31
INITW1  108-14#  108-26
INITW2  108-17#  108-23
```

CROSS REFERENCE TABLE (CREF V01-05)

```
INITW3 108-20   108-22#
INITW4 108-18   108-24#
INITWA  26-46#  108-12
INITWB  26-47#  108-21
INITWC  25-45#  108-32
INITXX 101-132  108-42#
INP28A  26-56#   64-39
INP28B  26-57#   64-43
INTB1M  26-28#   27-40
INTB2M  26-29#   27-41
INTB3M  26-30#   27-42
INTB4M  26-31#   27-45
INTBF   27-39#   85-37
INTBFL  27-45#   27-48
INTERR  27-69#  116-67
INTEST 112-45   116-1#
INTHD   25-9#    85-18    85-37    87-44    88-36    89-5
INTM1   26-22#   27-15
INTM2   26-23#   27-16    27-21
INTM3   26-24#   27-20
INTM4   26-25#   27-25
INTM5   26-26#   27-26
INTM6   26-27#   27-27
INTM7   26-75#   27-35
INTM8   26-76#   27-36
INTR1   27-14#   88-36
INTR2   27-19#   89-5
INTR3   27-24#   87-44
INTR4   27-30#   86-30   112-37
INTR5   27-34#   85-18
INTRCV  22-48#   96-2*   116-22*  116-31   116-43*  116-57
INTSRV  96-1#   116-17
INTST0  26-43#  116-16
INTST1  26-44#  116-69
INTST2  25-43#  116-67
INTST3  26-45#   27-70
INTST4  25-44#  116-41   116-61
IREST   22-41#  101-68   119-32
ISR      9-70#
ISTRT   22-42#  102-6    119-32
ISTRTH  22-43#   44-10   119-102
IXE      9-70#
J$JMP    5-16#   99-118
KNO    102-19   102-34#
KPRI   108-45#
KTBASA  22-36#   77-39*   78-14*   78-15*   79-17    80-18
KTBASE  78-12#
KTBASO  22-37#   77-38*   78-13*   79-13    80-14
KTMEM   23-10#   77-37*
KTREAD  79-12#
KTWRIT  80-13#
KW.BRL  23-4#   102-21*  102-24
KW.CSR  23-3#    34-33    34-47    75-29    88-28    95-8*    97-12    97-14*   98-13    99-60    101-74   102-17*  102-20*  102-25*
       115-12   116-34
KW.EL   23-7#    34-35    34-38    34-49    34-52    75-31    75-34    83-26    83-27    83-28    88-30    88-33    95-6*    95-7*
        98-15    98-16    99-62    99-63   101-80   101-83   102-11*  102-12*  102-30   102-31   115-14   115-17   116-36   116-39
KW.HZ   23-6#    83-16    98-17    99-64   102-23*  102-26
```

CROSS REFERENCE TABLE (CREF V01-05)

```
KW.VEC   23-5#    102-22*  102-24
KW11I    95-5#    102-24
KWOUT.   95-8      97-14   102-10#  102-25
KYES    102-14    102-16   102-20#
L$ACP     5-77#
L$APT     5-77#
L$AUT     5-77#
L$AUTO    5-77    110-10#
L$CCP     5-77#
L$CLEA    5-77    111-8#
L$CO      5-77#
L$DEPO    5-77#
L$DESC    5-77     24-22#
L$DESP    5-77#
L$DEVP    5-77#
L$DISP    5-77      6-8#
L$DLY     5-77#   101-87
L$DTP     5-77#
L$DTYP    5-77#
L$DUT     5-77#
L$DVTY    5-77     24-12#
L$EF      5-77#
L$ENVI    5-77#
L$ERRT    5-77     22-21#
L$ETP     5-77#
L$EXP1    5-77#
L$EXP4    5-77#
L$EXP5    5-77#
L$HARD    5-77    121-50   121-50#
L$HIME    5-77#    39-51
L$HPCP    5-77#
L$HPTP    5-77#
L$HW      5-77      7-10     7-10#
L$ICP     5-77#
L$INIT    5-77    101-8#
L$LADP    5-77#
L$LAST    5-77    125-14#  126-22
L$LOAD    5-77#
L$LUN     5-77#    33-23*   34-11*   59-29*   67-14*  112-23*
L$MREV    5-77#
L$NAME    5-77#
L$PRIO    5-77#
L$PROT    5-77    100-8#
L$PRT     5-77#
L$REPP    5-77#
L$REV     5-77#
L$RPT     5-77     99-45#
L$SOFT    5-77    123-12   123-12#
L$SPC     5-77#
L$SPCP    5-77#
L$SPTP    5-77#
L$STA     5-77#
L$SW      5-77      8-10     8-10#
L$TEST    5-77#
L$TIML    5-77#
L$UNIT    5-77#   101-130  105-16   108-4
L10000    7-10     7-27#
```

CROSS REFERENCE TABLE (CREF V01-05)

```
L10001    8-10      8-27#
L10002   27-17#
L10003   27-22#
L10004   27-28#
L10005   27-32#
L10006   27-37#
L10007   27-49#
L10010   27-54#
L10011   27-59#
L10012   27-63#
L10013   27-67#
L10014   27-71#
L10015   27-75#
L10016   27-79#
L10017   27-83#
L10020   67-48#
L10021   81-14#
L10022   82-22#
L10023   95-9#
L10024   96-3#
L10025   99-118    99-156#
L10027  108-45    109-15#
L10030  110-18#
L10031  111-30#
L10032  113-6#
L10033  112-40#
L10034  112-43#
L10035  112-46#
L10036  112-50#
L10037  112-56#
L10040  112-66#
L10041  117-24#
L10042  118-14#
L10043  119-117   120-189#
L10044  121-50    122-7#
L10045  123-12    124-11#
L10046  126-14#
L10050  126-14    126-21#
LBUFE    23-23#    51-50*    51-53
LBUFN    23-22#    51-47*    51-51     51-52*    51-53     51-62*   119-17
LBUFS    23-21#    31-15*    51-43     51-46*   119-10    119-13*
LDDM     33-22#    33-31
LDNEXT   33-24     33-27     33-29#
LOAD     69-23     70-15#
LOADB    68-43#    69-31
LOADDM   33-26     68-14#
LOADE1   68-56     70-26     71-3#
LOADER   68-37     68-53     69-24     70-23     71-4#
LOADM1   25-8#     71-3
LOADT1   68-41     69-4#
LOE       9-70#
LOG      11-32#
LOGCHK  119-11    119-22#
LOGDAT   26-77#    51-60
LOGFUL   26-78#    51-63
LOGM1    26-79#   119-12
LOGM2    26-80#   119-19
```

CROSS REFERENCE TABLE (CREF V01-05)

CROSS REFERENCE TABLE (CREF V01-05)

O$BGNR	5-16#	5-55#	5-77									
O$BGNS	5-16#	5-55#	5-77									
O$DU	5-16#	5-77										
O$ERRT	5-16#	5-55#	5-77									
O$GNSW	5-16#	5-55#	5-77									
O$POIN	5-16#	5-55	5-55#	5-55#	5-55#	5-55#	5-55#	5-77				
O$SETU	5-16#	5-55#	5-77	125-14								
ONEFIL	1-4#	1-8	3-:66	4-1	5-7#	5-37	9-10	98-30	99-10	112-11	120-192	121-11
OP.ABO	13-3#											
OP.ACC	13-4#											
OP.AVA	13-22#											
OP.AVL	13-5#											
OP.CCD	13-6#											
OP.CMP	13-7#											
OP.DUP	13-23#											
OP.ELP	13-29#											
OP.END	13-20#	35-5	35-8									
OP.ERS	13-8#											
OP.ESP	13-28#	68-43										
OP.FLU	13-9#											
OP.GCS	13-10#											
OP.GSS	13-27#											
OP.GUS	13-11#											
OP.MRD	13-18#											
OP.MWR	13-19#	70-16	72-21									
OP.ONL	13-12#											
OP.RD	13-13#											
OP.RLC	13-25#											
OP.RPL	13-14#											
OP.RSD	13-31#	35-8	36-28									
OP.SCC	13-15#											
OP.SEX	13-21#											
OP.SHC	13-24#											
OP.SSD	13-30#	35-5	36-9									
OP.SUC	13-16#											
OP.WR	13-17#											
OSTRE	55-12	55-19	55-24#									
OSTRNG	53-30	55-11#	55-23	57-9	67-47							
P.BCNT	15-21#	16-9#	68-46*	70-19*	74-19*							
P.BUFF	15-22#											
P.CMST	16-14#											
P.CNCL	16-48#											
P.CNTF	15-40#	16-46#										
P.CNTI	16-49#											
P.CPSP	15-34#											
P.CRF	15-17#	16-4#	35-23	73-17*								
P.CTMO	16-47#											
P.CYL	16-26#											
P.DMDT	15-50#											
P.ELGF	15-32#											
P.FBBK	16-10#											
P.FLGS	16-7#											
P.GRP	16-25#											
P.HSTI	15-31#	16-19#	16-35#									
P.HTMO	15-41#											
P.LBN	15-24#											
P.MEDI	16-21#	16-37#										

CROSS REFERENCE TABLE (CREF V01-05)

```
P.MLUN  16-17#   16-33#
P.MOD   15-20#
P.OPCD  15-19#   16-6#    35-9     72-29*
P.OTRF  15-27#   16-13#
P.OVRL  15-51#   68-47*   68-48*
P.RBN   15-36#
P.RBNS  16-28#
P.RCTC  16-29#
P.RCTS  16-27#
P.RGID  15-46#   70-21*
P.RGOF  15-47#   70-20*
P.SHST  16-23#   16-39#
P.SHUN  15-33#   16-22#   16-38#
P.STS   16-8#    35-16    68-55    70-25
P.TIME  15-43#
P.TRCK  16-24#
P.UADR  15-23#   68-45*   70-18*   74-18*
P.UNCL  16-40#
P.UNFL  15-30#   16-18#   16-34#
P.UNIT  15-18#   16-5#
P.UNSZ  16-41#
P.UNTI  16-20#   16-36#
P.USEF  15-42#
P.VRSN  15-39#   16-45#
P.VSER  16-42#
PAR0    20-3#    77-28*
PAR1    20-4#    77-29*
PAR2    20-5#    77-30*
PAR3    20-6#    77-31*
PAR4    20-7#    77-32*
PAR5    20-8#    77-33*
PAR6    20-9#    77-34*   79-20*   80-21*
PAR7    20-10#   77-35*
PAT16C  22-60#   42-19    119-64   119-65
PAT16W  22-61#  119-66
PDR0    20-12#   77-20*
PDR1    20-13#   77-21*
PDR2    20-14#   77-22*
PDR3    20-15#   77-23*
PDR4    20-16#   77-24*
PDR5    20-17#   77-25*
PDR6    20-18#   77-26*
PDR7    20-19#   77-27*
PNT      9-70#
PNTERR  51-40    67-11#   119-15
PNTNUM  56-25    56-31    56-37    61-14#
PNTNUS  58-12    61-16#
PORTST 112-42   114-1#
POWDLC 101-79#  101-82   101-84
POWDLL 101-87#
POWDLY 101-75   101-86#  101-90
PRI      9-70#
PRI00    9-70#  108-42   116-17   116-18   116-63
PRI01    9-70#
PRI02    9-70#
PRI03    9-70#
PRI04    9-70#
```

CROSS REFERENCE TABLE (CREF V01-05)

```
PRIO5    9-70#
PRIO6    9-70#
PRIO7    5-77     9-70#   68-32    77-17    86-24    112-30   110-44
RDDAT   93-34    93-36    93-54#
RDDLL   40-50    91-11#
RDERR   93-37    93-107#
RDREC   32-14    91-12    93-18#
RDST    93-20    93-24#   93-28    93-48    93-50    93-52    93-86    93-94
READDM  31-25    32-13#
README  32-15    32-20#
RESET   31-14    97-10#  101-71   112-27   117-10
RESETX  97-13    97-15#
RESPCT  34-8#    34-63
RESPDM  34-6#    34-64    40-62   112-63   117-16   118-12   119-27   119-110
RG.FLG  11-4#    73-24
RG.OWN  11-3#    73-24    73-25
RNTIME  67-38    98-12#
RNTIMX  98-14    98-24#
RPTCT   99-77#   99-116
RPTCTN  99-78    99-84    99-114#
RPTDT   99-83#   99-113
RPTDTN  99-86    99-111#
RPTEX   99-61    99-71#
RPTMD1  99-107   99-152#
RPTMD2  99-108   99-153#
RPTMH2  99-75    99-150#
RPTMS2  99-70    99-148#
RPTMSD  99-88    99-151#
RPTMSG  99-59    99-147#
RPTMSH  99-74    99-149#
RPTXX   99-73    99-117#
RSPDRP  34-23    34-41    34-68#   35-12    35-19    35-26    36-7     36-19
RSPDSP  36-18    36-46#   36-62
RSPIN   34-13    35-5#
RSPMUR  35-7     35-^#
RSPNRP  34-48    34  1    34-53    34-61#
RSPNTO  34-34    34-37    34-39    34-42#
RSPNXT  34-10    34-47#   34-70    36-42
RSPOU   34-15    35-31#
RSPOU2  36-26    36-35#
RSPOU3  36-33    36-36#
RSPOUT  35-31    36-24#
RSPPKE  25-4#    35-11
RSPPNE  25-6#    36-6
RSPPRE  25-5#    35-25
RSPPT2  36-3#
RSPPT3  36-5     36-9#
RSPPTW  35-24    35-30#
RSPREF  35-17    35-23#
RSPRPT  34-50    34-54#
RSPSTF  25-7#    35-18
RSPSTS  35-10    35-16#
RSPTM   34-21    34-27#
RSPTME  27-81#   34-22    75-44
RSPTMM  26-73#   27-82
RSPTMO  34-36    34-40#
RSPTOE  27-77#   34-40    75-36
```

CROSS REFERENCE TABLE (CREF V01-05)

```
RSPTOM  26-72#  27-78
RUNDM   33-15#  40-60   112-61   117-14   118-10   119-108
RWORDT  93-58   93-89#
RWRDE1  93-30   94-1#
RWRDE3  93-44   93-87    93-92    94-3#
RWRDEM  25-10#  94-1     94-3
RWRDF1  26-32#  27-52    27-57
RWRDF2  26-33#  27-53
RWRDF4  26-34#  27-58
RWRDM1  27-51#  94-1
RWRDM3  27-56#  94-3
S$LSYM   5-16#   7-27#    8-27#   27-17#   27-22#   27-28#   27-32#   27-37#   27-49#   27-54#   27-59#   27-63#   27-67#   27-71#
        27-75#  27-79#   27-83#   41-66    41-66    41-66    41-66#   67-48#   81-14#   82-22#   95-9#    96-3#    99-156#  108-32
       108-32  108-32   108-32#  109-15#  110-18#  111-30#  112-40#  112-43#  112-46#  112-50#  112-56#  112-66#  113-6#   117-24#
       118-14# 119-64   119-64   119-64   119-64#  119-68   119-68   119-68   119-68#  120-14   120-14   120-14   120-14#  120-23
       120-23  120-23   120-23#  120-29   120-29   120-29   120-29#  120-32   120-32   120-32   120-32#  120-33   120-33   120-33
       120-33# 120-36   120-36   120-36   120-36#  120-41   120-41   120-41   120-41#  120-50   120-50   120-50   120-50#  120-51
       120-51  120-51   120-51#  120-54   120-54   120-54   120-54#  120-57   120-57   120-57   120-57#  120-59   120-59   120-59
       120-59# 120-83   120-83   120-83   120-83#  120-121  120-121  120-121  120-121# 120-127  120-127  120-127  120-127# 120-131
       120-131 120-131  120-131# 120-146  120-146  120-146  120-146# 120-150  120-150  120-150  120-150# 120-156  120-156  120-156
       120-156# 120-160 120-160  120-160  120-160# 120-166  120-166  120-166  120-166# 120-180  120-180  120-180  120-180# 120-184
       120-184 120-184  120-184# 120-189# 122-7#   124-11#
S.EL    124-3   124-23#
S.IW    124-6   124-26#
S.LOG   124-7   124-27#
S.MAN   124-1   124-20#
S.MES   124-2   124-21#
S.SSF   124-5   124-25#
S.XL    124-4   124-24#
SA.A2    10-18#
SA.BST   10-65#
SA.CMD   10-27#
SA.CME   10-34#
SA.CTP   10-37#
SA.DI    10-19#  87-14
SA.ERC   10-13#
SA.ERR   10-9#   88-19
SA.GO    10-63#  85-45
SA.INE   10-47#
SA.INT   10-25#
SA.LFC   10-64#
SA.MCV   10-58#  87-38
SA.MSE   10-33#
SA.MSG   10-26#
SA.NV    10-17#
SA.NVE   10-48#
SA.PRG   10-41#
SA.S1    10-5#   86-36   87-14   114-8   116-24
SA.S2    10-6#   87-23
SA.S3    10-7#   87-31
SA.S4    10-8#   86-46   87-38
SA.STE   10-36#
SA.STP   10-29#  86-15
SA.TST   10-54#  87-8
SA.VCE   10-46#
SA.VEC   10-24#
SAMTAB  103-21   105-3#
```

```
SEKERE   48-14    48-17#
SETOO    83-17#   83-22
SETO1    83-18    83-20#
SETO2    83-26#   83-29
SETTO    34-57    36-41    75-15    83-14#   88-23    101-78   115-8    116-30
SFPTBL    8-10#   41-29    44-12    49-24    49-26     51-33    51-38    52-39   119-22   119-34   119-111
SM.IW    44-12    123-30#  124-6
SM.LOG   51-38    119-22   119-111  123-29#  124-7
SM.MAN   41-29    119-34   123-27#  124-1
SM.SSF   51-33    123-28#  124-5
SNDC1    73-21#   73-23
SNDCMD   36-37    68-52    70-22    73-14#
SO.BIT   41-29    44-12    51-33    51-38    119-22   119-34   119-111  123-26#  124-1    124-1    124-1    124-5    124-5    124-5
        124-6    124-6    124-6    124-7    124-7    124-7
SO.EL    52-39    123-24#  124-3    124-3    124-3
SO.XL    49-24    49-26    123-25#  124-4    124-4    124-4
SRO      20-28#   77-18*   79-23*   79-25*   80-24*   80-26*
ST.ABO   17-7#
ST.AVL   17-9#
ST.CMD   17-6#
ST.CMP   17-12#
ST.CNT   17-15#
ST.DAT   17-13#
ST.DIA   17-17#
ST.DRV   17-16#
ST.HST   17-14#
ST.MFE   17-10#
ST.MSK   17-3#    35-16    68-55    70-25
ST.OFL   17-8#
ST.SUB   17-4#
ST.SUC   17-5#
ST.WPR   17-11#
STIME    23-12#   34-49    34-52    34-56    101-77   101-80   101-83   102-31*  102-32*  102-33*
SVCGBL    5-16#    5-25#    5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     6-8      7-10
          7-10     8-10     8-10    22-21    24-12    24-22    27-11#   27-14    27-14    27-14    27-19    27-19    27-19    27-24
         27-24    27-24    27-30    27-30    27-30    27-34    27-34    27-34    27-39    27-39    27-39    27-51    27-51    27-51
         27-56    27-56    27-56    27-61    27-61    27-61    27-65    27-65    27-65    27-69    27-69    27-69    27-73    27-73
         27-73    27-77    27-77    27-77    27-81    27-81    27-81    27-88#   67-31    81-10    82-18    95-5     96-1     99-45
        100-8    101-8    110-10   111-8    121-50   123-12   125-14   125-14   125-14   125-14#
SVCINS    5-16#    5-22#    5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77     5-77
          5-77     5-77     5-77     6-8      6-8      6-8      6-8      6-8      6-8      6-8      6-8      6-8      6-8      6-8
          6-8      6-8      6-8      6-8      7-10     7-10     7-10     8-10     8-10     8-10    24-12    24-12    24-12    24-12
         24-12    24-12    24-22    24-22    24-22    24-22    24-22    24-22    27-8#    27-15    27-15    27-15    27-15    27-15
         27-16    27-16    27-16    27-16    27-16    27-16    27-17    27-20    27-20    27-20    27-20    27-20    27-21    27-21
         27-21    27-21    27-21    27-21    27-22    27-25    27-25    27-25    27-25    27-25    27-26    27-26    27-26    27-26
         27-26    27-26    27-27    27-27    27-27    27-27    27-27    27-27    27-28    27-31    27-31    27-31    27-31    27-31
```

27-31	27-32	27-35	27-35	27-35	27-35	27-35	27-36	27-36	27-36	27-36	27-36	27-36	27-37
27-40	27-40	27-40	27-40	27-40	27-41	27-41	27-41	27-41	27-41	27-42	27-42	27-42	27-42
27-42	27-45	27-45	27-45	27-45	27-45	27-45	27-45	27-49	27-52	27-52	27-52	27-52	27-52
27-52	27-53	27-53	27-53	27-53	27-53	27-54	27-57	27-57	27-57	27-57	27-57	27-57	27-58
27-58	27-58	27-58	27-58	27-59	27-62	27-62	27-62	27-62	27-62	27-62	27-62	27-63	27-66
27-66	27-66	27-66	27-66	27-66	27-66	27-67	27-70	27-70	27-70	27-70	27-70	27-70	27-71
27-74	27-74	27-74	27-74	27-74	27-74	27-75	27-78	27-78	27-78	27-78	27-78	27-78	27-79
27-82	27-82	27-82	27-82	27-82	27-82	27-82	27-83	27-85#	28-7	28-7	28-7	28-7	28-7
28-7	28-7	28-7	28-7	28-7	28-7	28-7	28-8	28-8	28-8	32-20	32-20	32-20	32-20
32-20	32-20	32-20	32-20	32-20	32-20	32-20	32-20	32-21	32-21	32-21	34-22	34-22	34-22
34-22	34-22	34-22	34-22	34-22	34-22	34-22	34-22	34-22	34-40	34-40	34-40	34-40	34-40
34-40	34-40	34-40	34-40	34-40	34-40	34-40	34-43	34-43	34-43	34-54	34-54	34-54	35-11
35-11	35-11	35-11	35-11	35-11	35-11	35-11	35-11	35-11	35-11	35-11	35-18	35-18	35-18
35-18	35-18	35-18	35-18	35-18	35-18	35-18	35-18	35-18	35-25	35-25	35-25	35-25	35-25
35-25	35-25	35-25	35-25	35-25	35-25	35-25	36-6	36-6	36-6	36-6	36-6	36-6	36-6
36-6	36-6	36-6	36-6	36-6	41-32	41-32	41-32	41-33	41-33	41-33	41-34	41-34	41-34
41-34	41-34	41-34	41-34	41-34	41-34	41-34	41-34	41-34	41-34	41-34	41-34	41-42	41-42
41-42	41-42	41-42	41-42	41-42	41-42	41-42	41-42	41-42	41-42	41-42	41-42	41-42	41-42
41-42	41-42	41-42	41-42	41-42	41-42	41-42	41-42	41-43	41-43	41-43	41-43	41-43	41-43
41-43	41-43	41-43	41-43	41-43	41-43	41-43	41-43	41-43	41-44	41-44	41-44	41-44	41-44
41-44	41-44	41-44	41-44	41-44	41-44	41-44	41-44	41-44	41-44	41-45	41-45	41-45	41-45
41-45	41-45	41-45	41-45	41-45	41-45	41-45	41-45	41-45	41-45	41-45	41-53	41-53	41-53
41-53	41-53	41-53	41-53	41-53	41-53	41-53	41-53	41-53	41-53	41-53	41-53	41-53	41-53
41-53	41-59	41-59	41-59	41-59	41-59	41-59	41-59	41-59	41-59	41-59	41-59	41-59	41-59
41-59	41-59	41-59	41-59	41-59	41-62	41-62	41-62	41-62	41-62	41-62	41-62	41-62	41-62
41-62	41-62	41-62	41-62	41-62	41-62	41-66	41-66	41-66	41-66	41-66	41-66	41-66	41-66
41-66	41-66	41-66	41-66	41-66	41-66	41-66	41-66	41-66	41-66	41-66	41-66	41-66	41-66
41-66	41-66	41-113	41-113	41-113	41-113	41-113	41-113	41-113	41-113	41-113	41-113	41-113	41-113
41-113	41-113	41-113	49-28	49-28	49-28	49-33	49-33	49-33	49-33	49-33	49-33	49-33	49-33
49-33	49-33	49-33	49-33	49-33	49-33	49-33	49-33	49-33	49-33	49-33	49-33	49-33	49-33
49-33	49-33	49-34	49-34	49-34	49-34	49-34	49-34	49-34	49-34	49-34	49-34	49-34	49-34
49-34	49-34	49-34	51-27	51-27	51-27	51-60	51-60	51-60	51-60	51-60	51-60	51-60	51-60
51-60	51-60	51-60	51-60	51-60	51-60	51-60	51-63	51-63	51-63	51-63	51-63	51-63	51-63
51-63	51-63	51-63	51-63	51-63	51-63	51-63	51-63	52-41	52-41	52-41	52-44	52-44	52-44
52-44	52-44	52-4,	52-44	52-44	52-44	52-44	52-44	52-44	52-44	52-44	52-44	52-44	52-44
52-44	53-21	53-21	53-21	53-21	53-21	53-21	53-21	53-21	53-21	53-21	53-21	53-21	53-21
53-21	53-21	53-21	53-21	53-21	53-21	53-21	53-21	53-21	53-21	53-21	55-18	55-18	55-18
55-18	55-18	55-18	55-18	55-18	55-18	55-18	55-18	55-18	55-18	55-18	55-18	56-6	56-6
56-6	56-6	56-6	56-6	56-6	56-6	56-6	56-6	56-6	56-6	56-6	56-6	56-6	56-6
56-6	56-6	56-14	56-14	56-14	56-14	56-14	56-14	56-14	56-14	56-14	56-14	56-14	56-14
56-14	56-14	56-14	56-14	56-14	56-14	56-43	56-43	56-43	56-43	56-43	56-43	56-43	56-43
56-43	56-43	56-43	56-43	56-43	56-43	56-43	56-56	56-56	56-56	56-56	56-56	56-56	56-56
56-56	56-56	56-56	56-56	56-56	56-56	56-56	56-56	56-64	56-64	56-64	56-64	56-64	56-64
56-64	56-64	56-64	56-64	56-64	56-64	56-64	56-64	56-64	56-64	56-64	56-64	58-13	58-13
58-13	58-13	58-13	58-13	58-13	58-13	58-13	58-13	58-13	58-13	58-13	58-13	58-13	58-20
58-20	58-20	58-20	58-20	58-20	58-20	58-20	58-20	58-20	58-20	58-20	58-20	58-20	58-20
59-24	59-24	59-24	59-24	59-24	59-24	59-24	59-24	59-24	59-24	59-24	59-24	61-59	61-59
61-59	61-59	61-59	61-59	61-59	61-59	61-59	61-59	61-59	61-59	61-59	61-59	61-59	61-59
61-59	61-59	61-69	61-69	61-69	61-69	61-69	61-69	61-69	61-69	61-69	61-69	61-69	61-69
61-69	61-69	61-69	61-69	61-69	61-69	64-39	64-39	64-39	64-39	64-39	64-39	64-39	64-39
64-39	64-39	64-39	64-39	64-39	64-39	64-39	64-43	64-43	64-43	64-43	64-43	64-43	64-43
64-43	64-43	64-43	64-43	64-43	64-43	64-43	64-43	65-4	65-4	65-4	65-4	65-4	65-4
65-4	65-4	65-4	65-4	65-4	65-4	65-4	65-4	65-4	65-4	65-4	65-4	65-13	65-13
65-13	65-13	65-13	65-13	65-13	65-13	65-13	65-13	65-13	65-13	65-13	65-13	65-13	65-13
65-13	65-13	65-30	65-30	65-30	65-30	65-30	65-30	65-30	65-30	65-30	65-30	65-30	65-30
65-30	65-30	65-30	65-30	65-30	65-30	67-27	67-27	67-27	67-34	67-34	67-34	67-34	67-34
67-34	67-34	67-34	67-34	67-34	67-34	67-34	67-34	67-34	67-34	67-34	67-34	67-34	67-34

67-34	67-34	67-34	67-34	67-34	67-37	67-37	67-37	67-37	67-37	67-37	67-37	67-37	67-37
67-37	67-37	67-37	67-37	67-37	67-37	67-37	67-37	67-37	67-48	67-48	67-48	68-32	68-32
68-32	68-32	68-32	68-32	68-32	68-32	68-32	68-32	68-32	68-32	68-32	68-32	68-32	68-32
68-32	68-32	71-3	71-3	71-3	71-3	71-3	71-3	71-3	71-3	71-3	71-3	71-3	71-3
75-28	75-28	75-28	75-36	75-36	75-36	75-36	75-36	75-36	75-36	75-36	75-36	75-36	75-36
75-36	75-44	75-44	75-44	75-44	75-44	75-44	75-44	75-44	75-44	75-44	75-44	75-44	77-17
77-17	77-17	77-17	77-17	77-17	77-17	77-17	77-17	77-17	77-17	77-17	77-17	77-17	77-17
77-17	77-17	77-17	77-36	77-36	77-36	77-36	77-36	77-36	81-14	81-14	81-14	82-22	82-22
82-22	85-18	85-18	85-18	85-18	85-18	85-18	85-18	85-18	85-18	85-18	85-18	85-18	85-37
85-37	85-37	85-37	85-37	85-37	85-37	85-37	85-37	85-37	85-37	85-37	86-13	86-13	86-13
86-24	86-24	86-24	86-24	86-24	86-24	86-24	86-24	86-24	86-24	86-24	86-24	86-24	86-24
86-24	86-24	86-24	86-24	86-27	86-27	86-27	86-27	86-27	86-27	86-30	86-30	86-30	86-30
86-30	86-30	86-30	86-30	86-30	86-30	86-30	86-30	87-44	87-44	87-44	87-44	87-44	87-44
87-44	87-44	87-44	87-44	87-44	87-44	88-27	88-27	88-27	88-36	88-36	88-36	88-36	88-36
88-36	88-36	88-36	88-36	88-36	88-36	88-36	89-5	89-5	89-5	89-5	89-5	89-5	89-5
89-5	89-5	89-5	89-5	89-5	92-12	92-12	92-12	93-21	93-21	93-21	93-21	93-21	93-21
93-24	93-24	93-24	93-26	93-26	93-26	93-26	93-26	93-26	93-31	93-31	93-31	93-41	93-41
93-41	93-76	93-76	93-76	93-83	93-83	93-83	93-89	93-89	93-89	93-99	93-99	93-99	93-102
93-102	93-102	94-1	94-1	94-1	94-1	94-1	94-1	94-1	94-1	94-1	94-1	94-1	94-1
94-2	94-2	94-2	94-3	94-3	94-3	94-3	94-3	94-3	94-3	94-3	94-3	94-3	94-3
94-3	94-4	94-4	94-4	95-9	95-9	95-9	96-3	96-3	96-3	97-10	97-10	97-10	97-11
97-11	97-11	98-23	98-23	98-23	98-23	98-23	98-23	98-23	98-23	98-23	98-23	98-23	98-23
98-23	98-23	98-23	98-23	98-23	98-23	98-23	98-23	98-23	98-23	98-23	98-23	98-24	98-24
98-24	98-24	98-24	98-24	98-24	98-24	98-24	98-24	98-24	98-24	98-24	98-24	98-24	99-59
99-59	99-59	99-59	99-59	99-59	99-59	99-59	99-59	99-59	99-59	99-59	99-59	99-59	99-59
99-59	99-59	99-59	99-70	99-70	99-70	99-70	99-70	99-70	99-70	99-70	99-70	99-70	99-70
99-70	99-70	99-70	99-70	99-70	99-70	99-70	99-70	99-70	99-70	99-70	99-70	99-70	99-71
99-71	99-71	99-71	99-71	99-71	99-71	99-71	99-71	99-71	99-71	99-71	99-71	99-71	99-71
99-74	99-74	99-74	99-74	99-74	99-74	99-74	99-74	99-74	99-74	99-74	99-74	99-74	99-74
99-74	99-75	99-75	99-75	99-75	99-75	99-75	99-75	99-75	99-75	99-75	99-75	99-75	99-75
99-75	99-75	99-88	99-88	99-88	99-88	99-88	99-88	99-88	99-88	99-88	99-88	99-88	99-88
99-88	99-88	99-88	99-88	99-88	99-88	99-88	99-88	99-88	99-107	99-107	99-107	99-107	99-107
99-107	99-107	99-107	99-107	99-107	99-107	99-107	99-107	99-107	99-107	99-107	99-107	99-107	99-107
99-107	99-107	99-107	99-107	99-107	99-107	99-107	99-107	99-108	99-108	99-108	99-108	99-108	99-108
99-108	99-108	99-108	99-108	99-108	99-108	99-108	99-108	99-108	99-108	99-108	99-108	99-108	99-108
99-108	99-108	99-108	99-108	99-118	99-118	99-118	99-118	99-118	99-118	99-156	99-156	99-156	101-63
101-63	101-63	101-63	101-63	101-63	101-64	101-64	101-64	101-66	101-66	101-66	101-66	101-66	101-66
101-67	101-67	101-67	101-69	101-69	101-69	101-69	101-69	101-69	101-70	101-70	101-70	101-73	101-73
101-73	101-79	101-79	101-79	101-87	101-87	101-87	101-87	101-87	101-87	101-87	101-87	101-87	101-87
101-87	101-87	101-87	101-87	101-87	101-87	101-87	101-87	101-87	101-87	101-87	101-87	101-87	101-87
101-88	101-88	101-88	101-91	101-91	101-91	101-91	101-91	101-91	101-92	101-92	101-92	101-114	101-114
101-114	101-114	101-114	101-114	101-115	101-115	101-115	101-122	101-122	101-122	101-122	101-122	101-122	101-122
101-122	101-122	101-122	101-122	101-122	101-123	101-123	101-123	102-1	102-1	102-1	102-3	102-3	102-3
102-3	102-3	102-3	102-13	102-13	102-13	102-13	102-13	102-13	102-14	102-14	102-14	102-15	102-15
102-15	102-15	102-15	102-15	102-16	102-16	102-16	102-18	102-18	102-18	102-18	102-18	102-18	102-18
102-18	102-18	102-18	102-18	102-18	102-18	102-18	102-18	102-24	102-24	102-24	102-24	102-24	102-24
102-24	102-24	102-24	102-24	102-24	102-24	102-24	102-24	102-24	102-24	102-24	102-24	103-10	103-10
103-10	103-10	103-10	103-10	103-11	103-11	103-11	106-5	106-5	106-5	106-5	106-5	106-5	106-6
106-6	106-6	108-12	108-12	108-12	108-12	108-12	108-12	108-12	108-12	108-12	108-12	108-12	108-12
108-12	108-12	108-12	108-21	108-21	108-21	108-21	108-21	108-21	108-21	108-21	108-21	108-21	108-21
108-21	108-21	108-21	108-21	108-21	108-21	108-21	108-21	108-21	108-21	108-21	108-21	108-21	108-30
108-30	108-30	108-31	108-31	108-31	108-32	108-32	108-32	108-32	108-32	108-32	108-32	108-32	108-32
108-32	108-32	108-32	108-32	108-32	108-32	108-32	108-32	108-32	108-35	108-35	108-35	108-42	108-42
108-42	108-42	108-42	108-42	108-45	108-45	108-45	108-45	108-45	108-45	109-2	109-2	109-2	109-2
109-2	109-2	109-2	109-2	109-2	109-2	109-2	109-2	109-3	109-3	109-3	109-6	109-6	109-6
109-6	109-6	109-6	109-6	109-6	109-6	109-6	109-6	109-6	109-7	109-7	109-7	109-11	109-11
109-11	109-11	109-11	109-11	109-11	109-11	109-11	109-11	109-11	109-11	109-12	109-12	109-12	109-15

CROSS REFERENCE TABLE (CREF V01-05)

109-15	109-15	110-18	110-18	110-18	111-30	111-30	111-30	112-28	112-28	112-28	112-30	112-30	112-30
112-30	112-30	112-30	112-30	112-30	112-30	112-30	112-30	112-30	112-30	112-30	112-30	112-30	112-30
112-30	112-34	112-34	112-34	112-34	112-34	112-34	112-37	112-37	112-37	112-37	112-37	112-37	112-37
112-37	112-37	112-37	112-37	112-37	112-40	112-40	112-40	112-41	112-41	112-41	112-43	112-43	112-43
112-44	112-44	112-44	112-46	112-46	112-46	112-47	112-47	112-47	112-50	112-50	112-50	112-51	112-51
112-51	112-56	112-56	112-56	112-57	112-57	112-57	112-66	112-66	112-66	113-6	113-6	113-6	1.4-17
114-17	114-17	114-17	114-17	114-17	114-17	114-17	114-17	114-17	114-17	114-17	114-27	114-27	114-27
114-27	114-27	114-27	114-27	114-27	114-27	114-27	114-27	114-27	115-11	115-11	115-11	115-19	115-19
115-19	115-19	115-19	115-19	115-19	115-19	115-19	115-19	115-19	115-19	116-16	116-16	116-16	116-16
116-16	116-16	116-16	116-16	116-16	116-16	116-16	116-16	116-16	116-16	116-16	116-16	116-16	116-16
116-16	116-16	116-16	116-17	116-17	116-17	116-17	116-17	116-17	116-17	116-17	116-17	116-17	116-17
116-17	116-17	116-17	116-17	116-17	116-17	116-17	116-18	116-18	116-18	116-18	116-18	116-18	116-33
116-33	116-33	116-41	116-41	116-41	116-41	116-41	116-41	116-41	116-41	116-41	116-41	116-41	116-41
116-44	116-44	116-44	116-44	116-44	116-44	116-55	116-55	116-55	116-55	116-55	116-55	116-61	116-61
116-61	116-61	116-61	116-61	116-61	116-61	116-61	116-61	116-61	116-61	116-63	116-63	116-63	116-63
116-63	116-63	116-67	116-67	116-67	116-67	116-67	116-67	116-67	116-67	116-67	116-67	116-67	116-67
116-69	116-69	116-69	116-69	116-69	116-69	116-69	116-69	116-69	116-69	116-69	116-69	116-69	116-69
116-69	116-72	116-72	116-72	116-72	116-72	116-72	116-77	116-77	116-77	116-77	116-77	116-77	117-24
117-24	117-24	118-14	118-14	118-14	119-12	119-12	119-12	119-12	119-12	119-12	119-12	119-12	119-12
119-12	119-12	119-12	119-12	119-12	119-12	119-19	119-19	119-19	119-19	119-19	119-19	119-19	119-19
119-19	119-19	119-19	119-19	119-19	119-19	119-19	119-24	119-24	119-24	119-24	119-24	119-24	119-24
119-24	119-24	119-24	119-24	119-24	119-24	119-24	119-24	119-36	119-36	119-36	119-37	119-37	119-37
119-64	119-64	119-64	119-64	119-64	119-64	119-64	119-64	119-64	119-64	119-64	119-64	119-64	119-64
119-64	119-64	119-64	119-64	119-64	119-64	119-64	119-64	119-64	119-64	119-68	119-68	119-68	119-68
119-68	119-68	119-68	119-68	119-68	119-68	119-68	119-68	119-68	119-68	119-68	119-68	119-68	119-68
119-68	119-68	119-68	119-68	119-68	119-68	119-76	119-76	119-76	119-76	119-76	119-76	119-76	119-76
119-76	119-76	119-76	119-76	119-76	119-76	119-76	119-113	119-113	119-113	119-116	119-116	119-116	119-117
119-117	119-117	119-117	119-117	119-117	120-12	120-12	120-12	120-12	120-12	120-12	120-12	120-12	120-12
120-12	120-12	120-12	120-12	120-12	120-12	120-12	120-12	120-12	120-12	120-12	120-12	120-12	120-12
120-12	120-14	120-14	120-14	120-14	120-14	120-14	120-14	120-14	120-14	120-14	120-14	120-14	120-14
120-14	120-14	120-14	120-14	120-14	120-14	120-14	120-14	120-14	120-14	120-14	120-23	120-23	120-23
120-23	120-23	120-23	120-23	120-23	120-23	120-23	120-23	120-23	120-23	120-23	120-23	120-23	120-23
120-23	120-23	120-23	120-23	120-23	120-23	120-23	120-29	120-29	120-29	120-29	120-29	120-29	120-29
120-29	120-29	120-29	120-29	120-29	120-29	120-29	120-29	120-29	120-29	120-29	120-32	120-32	120-32
120-32	120-32	120-32	120-32	120-32	120-32	120-32	120-32	120-32	120-32	120-32	120-32	120-32	120-32
120-32	120-33	120-33	120-33	120-33	120-33	120-33	120-33	120-33	120-33	120-33	120-33	120-33	120-33
120-33	120-33	120-33	120-33	120-33	120-36	120-36	120-36	120-36	120-36	120-36	120-36	120-36	120-36
120-36	120-36	120-36	120-36	120-36	120-36	120-36	120-36	120-36	120-41	120-41	120-41	120-41	120-41
120-41	120-41	120-41	120-41	120-41	120-41	120-41	120-41	120-41	120-41	120-41	120-41	120-41	120-41
120-41	120-41	120-41	120-41	120-41	120-50	120-50	120-50	120-50	120-50	120-50	120-50	120-50	120-50
120-50	120-50	120-50	120-50	120-50	120-50	120-50	120-50	120-50	120-51	120-51	120-51	120-51	120-51
120-51	120-51	120-51	120-51	120-51	120-51	120-51	120-51	120-51	120-51	120-51	120-51	120-51	120-54
120-54	120-54	120-54	120-54	120-54	120-54	120-54	120-54	120-54	120-54	120-54	120-54	120-54	120-54
120-54	120-54	120-54	120-57	120-57	120-57	120-57	120-57	120-57	120-57	120-57	120-57	120-57	120-57
120-57	120-57	120-57	120-57	120-57	120-57	120-57	120-59	120-59	120-59	120-59	120-59	120-59	120-59
120-59	120-59	120-59	120-59	120-59	120-59	120-59	120-59	120-59	120-59	120-59	120-77	120-77	120-77
120-77	120-77	120-77	120-77	120-77	120-77	120-77	120-77	120-77	120-77	120-77	120-77	120-78	120-78
120-78	120-78	120-78	120-78	120-78	120-78	120-78	120-78	120-78	120-78	120-78	120-78	120-78	120-79
120-79	120-79	120-79	120-79	120-79	120-79	120-79	120-79	120-79	120-79	120-79	120-79	120-79	120-79
120-80	120-80	120-80	120-80	120-80	120-80	120-80	120-80	120-80	120-80	120-80	120-80	120-80	120-80
120-80	120-81	120-81	120-81	120-81	120-81	120-81	120-81	120-81	120-81	120-81	120-81	120-81	120-81
120-81	120-81	120-82	120-82	120-82	120-82	120-82	120-82	120-82	120-82	120-82	120-82	120-82	120-82
120-82	120-82	120-82	120-83	120-83	120-83	120-83	120-83	120-83	120-83	120-83	120-83	120-83	120-83
120-83	120-83	120-83	120-83	120-83	120-83	120-83	120-83	120-83	120-83	120-83	120-83	120-83	120-121
120-121	120-121	120-121	120-121	120-121	120-121	120-121	120-121	120-121	120-121	120-121	120-121	120-121	120-121
120-121	120-121	120-121	120-121	120-121	120-121	120-121	120-121	120-121	120-127	120-127	120-127	120-127	120-127
120-127	120-127	120-127	120-127	120-127	120-127	120-127	120-127	120-127	120-127	120-127	120-127	120-127	120-127

```
        120-127 120-127 120-127 120-127 120-127 120-131 120-131 120-131 120-131 120-131 120-131 120-131 120-131 120-131
        120-131 120-131 120-131 120-131 120-131 120-131 120-131 120-131 120-131 120-131 120-131 120-131 120-131 120-131
        120-131 120-146 120-146 120-146 120-146 120-146 120-146 120-146 120-146 120-146 120-146 120-146 120-146 120-146
        120-146 120-146 120-146 120-146 120-146 120-146 120-146 120-146 120-146 120-146 120-146 120-150 120-150 120-150
        120-150 120-150 120-150 120-150 120-150 120-150 120-150 120-150 120-150 120-150 120-150 120-150 120-150 120-150
        120-150 120-150 120-150 120-150 120-150 120-150 120-150 120-156 120-156 120-156 120-156 120-156 120-156 120-156
        120-156 120-156 120-156 120-156 120-156 120-156 120-156 120-156 120-156 120-156 120-156 120-156 120-156 120-156
        120-156 120-156 120-156 120-160 120-160 120-160 120-160 120-160 120-160 120-160 120-160 120-160 120-160 120-160
        120-160 120-160 120-160 120-160 120-160 120-160 120-160 120-160 120-160 120-160 120-160 120-160 120-160 120-166
        120-166 120-166 120-166 120-166 120-166 120-166 120-166 120-166 120-166 120-166 120-166 120-166 120-166 120-166
        120-166 120-166 120-166 120-180 120-180 120-180 120-180 120-180 120-180 120-180 120-180 120-180 120-180 120-180
        120-180 120-180 120-180 120-180 120-180 120-180 120-180 120-180  20-180 120-180 120-180 120-180 120-180 120-184
        120-184 120-184 120-184 120-184 120-184 120-184 120-184 120-184 120-184 120-184 120-184 120-184 120-184 120-184
        120-184 120-184 120-184 120-184 120-184 120-184 120-184 120-184 120-184 120-189 120-189 120-189 121-50  121-50
        121-50  122-1   122-1   122-1   122-1   122-1   122-1   122-1   122-1   122-1   122-1   122-1   122-1   122-1   122-2
        122-2   122-2   122-2   122-2   122-2   122-2   122-2   122-2   122-2   122-2   122-2   122-3   122-3   122-3
        122-3   122-3   122-3   122-3   122-3   122-3   122-3   122-3   122-3   122-3   122-3   122-3   122-4   122-4
        122-4   122-4   122-4   122-4   122-4   122-4   122-4   122-4   122-4   122-4   122-4   122-4   122-4   122-5
        122-5   122-5   122-5   122-5   122-5   122-5   122-5   122-5   122-5   122-5   122-5   122-5   122-5   122-5
        122-6   122-6   122-6   122-6   122-6   122-6   122-6   122-6   122-6   122-7   122-7   122-7   123-12  123-12
        123-12  124-1   124-1   124-1   124-1   124-1   124-1   124-1   124-1   124-1   124-2   124-2   124-2   124-2
        124-2   124-2   124-3   124-3   124-3   124-3   124-3   124-3   124-3   124-3   124-3   124-3   124-3   124-3
        124-3   124-3   124-3   124-4   124-4   124-4   124-4   124-4   124-4   124-4   124-4   124-4   124-4   124-4
        124-4   124-4   124-4   124-4   124-5   124-5   124-5   124-5   124-5   124-5   124-5   124-5   124-5   124-6
        124-6   124-6   124-6   124-6   124-6   124-6   124-6   124-6   124-7   124-7   124-7   124-7   124-7   124-7
        124-7   124-7   124-7   124-11  124-11  124-11  125-14  125-14  125-14  125-14  125-14  125-14  125-14  125-14
        125-14  126-14  126-14  126-14  126-14  126-14  126-14
SVCSUB   5-16#   5-24#  27-10#  27-87#  112-28  112-41  112-44  112-47  112-51  112-57
SVCTAG   5-16#   5-26#   7-27    8-27   27-12#  27-17   27-17   27-17   27-22   27-22   27-22   27-28   27-28   27-28
        27-32   27-32   27-32   27-37   27-37   27-37   27-49   27-49   27-49   27-54   27-54   27-54   27-59   27-59
        27-59   27-63   27-63   27-63   27-67   27-67   27-67   27-71   27-71   27-71   27-75   27-75   27-75   27-79
        27-79   27-79   27-83   27-83   27-83   27-89#  41-66   67-48   81-14   82-22   95-9    96-3    99-156  108-32
        109-15  110-18  111-30  112-40  112-43  112-46  112-50  112-56  112-66  113-6   117-24  118-14  119-64  119-68
        120-14  120-23  120-29  120-32  120-33  120-36  120-41  120-50  120-51  120-54  120-57  120-59  120-83  120-121
        120-127 120-131 120-136 120-150 120-156 120-160 120-166 120-180 120-184 120-189 122-7   124-11  126-14  126-21
SVCTST   5-16#   5-23#   27-9#  27-86#  112-18  117-3   118-3   119-3
TSSAUT  110-10#  110-18
TSSCLE  111-8#   111-30
TSSDAT  126-14   126-14#  126-21
TSSMAR  121-50   121-50#  122-7
TSSHW    7-10     7-10#    7-27
TSSINI  101-8#   108-45   109-15
TSSMSG   27-14#  27-17   27-19#  27-22   27-24#  27-28   27-30#  27-32   27-34#  27-37   27-39#  27-49   27-51#  27-54
        27-56#  27-59   27-61#  27-63   27-65#  27-67   27-69#  27-71   27-73#  27-75   27-77#  27-79   27-81#  27-83
        67-31#  67-48
TSSPC   126-13#  126-22
TSSPRO  100-8#
TSSPTA  126-13#  126-14  126-14#
TSSRPT   99-45#   99-118  99-156
TSSSOF  123-12   123-12#  124-11
TSSSRV   81-10#   81-14   82-18#  82-22   95-5#   95-9    96-1#   96-3
TSSSUB  112-28#  112-40  112-41#  112-43  112-44#  112-46  112-47#  112-50  112-51#  112-56  112-57#  112-66
TSSSW    8-10     8-10#    8-27
TSSTES  112-18#  113-6   117-3#  117-24  118-3#  118-14  119-3#  119-117 120-189
TSARGC   5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77#   5-77#   5-77#
        5-77#   5-77#   5-77#   27-15   27-15   27-15#  27-16   27-16   27-16   27-16#  27-16#  27-20   27-20   27-20#
        27-21   27-21   27-21   27-21#  27-21#  27-25   27-25   27-25#  27-26   27-26   27-26   27-26#  27-26#  27-27
```

(ROSS REFERENCE TABLE (CREF V01-05)

```
        27-27    27-27    27-27#   27-27#   27-31    27-31    27-31    27-31#   27-31#   27-35    27-35    27-35#   27-36    27-36
        27-36    27-36#   27-36#   27-40    27-40    27-40#   27-41    27-41    27-41#   27-42    27-42    27-42#   27-45    27-45
        27-45    27-45    27-45#   27-45#   27-45#   27-52    27-52    27-52    27-52#   27-52#   27-53    27-53    27-53#   27-57
        27-57    27-57    27-57#   27-57#   27-58    27-58    27-58#   27-62    27-62    27-62    27-62    27-62#   27-62#   27-62#
        27-66    27-66    27-66    27-66    27-66#   27-66#   27-66#   27-70    27-70    27-70    27-70#   27-70#   27-74    27-74
        27-74    27-74#   27-74#   27-78    27-78    27-78    27-78#   27-78#   27-82    27-82    27-82    27-82    27-82#   27-82#
        27-82#   41-34    41-34    41-34#   41-42    41-42    41-42    41-42    41-42    41-42#   41-42#   41-42#   41-42#   41-43
        41-43    41-43#   41-44    41-44    41-44#   41-45    41-45    41-45#   41-53    41-53    41-53    41-53#   41-53#   41-59
        41-59    41-59    41-59#   41-59#   41-62    41-62    41-62#   41-113   41-113   41-113#  49-33    49-33    49-33    49-33
        49-33    49-33#   49-33#   49-33#   49-33#   49-34    49-34    49-34#   51-60    51-60    51-60#   51-63    51-63    51-63#
        52-44    52-44    52-44    52-44#   52-44#   53-21    53-21    53-21    53-21    53-21    53-21#   53-21#   53-21#   53-21#
        55-18    55-18    55-18#   56-6     56-6     56-6     56-6#    56-6#    56-14    56-14    56-14    56-14#   56-14#   56-43
        56-43    56-43#   56-56    56-56    56-56#   56-64    56-64    56-64    56-64#   56-64#   58-13    58-13    58-13#   58-20
        58-20    58-20#   61-59    61-59    61-59    61-59#   61-59#   61-69    61-69    61-69    61-69#   61-69#   64-39    64-39
        64-39#   64-43    64-43    64-43#   65-4     65-4     65-4     65-4#    65-4#    65-13    65-13    65-13    65-13#   65-13#
        65-30    65-30    65-30    65-30#   65-30#   67-34    67-34    67-34    67-34    67-34    67-34#   67-34#   67-34#   67-34#
        67-37    67-37    67-37    67-37#   67-37#   98-23    98-23    98-23    98-23    98-23    98-23#   98-23#   98-23#   98-23#
        98-24    98-24    98-24#   99-59    99-59    99-59    99-59#   99-59#   99-70    99-70    99-70    99-70    99-70    99-70#
        99-70#   99-70#   99-70#   99-71    99-71    99-71#   99-74    99-74    99-74#   99-75    99-75    99-75#   99-88    99-88
        99-88    99-88    99-88#   99-88#   99-88#   99-107   99-107   99-107   99-107   99-107   99-107   99-107#  99-107#  99-107#
        99-107#  99-107#  99-108   99-108   99-108   99-108   99-108   99-108#  99-108#  99-108#  99-108#  102-18   102-18   102-18#
       108-12   108-12   108-12#  108-21   108-21   108-21   108-21   108-21   108-21#  108-21#  108-21#  108-21#  116-16   116-16
       116-16   116-16   116-16#  116-16#  116-16#  116-69   116-69   116-69#  119-12   119-12   119-12#  119-19   119-19   119-19#
       119-24   119-24   119-24#  119-76   119-76   119-76#  120-12   120-12   120-12   120-12   120-12   120-12#  120-12#  120-12#
       120-12#  120-77   120-77   120-77#  120-78   120-78   120-78#  120-79   120-79   120-79#  120-80   120-80   120-80#  120-81
       120-81   120-81#  120-82   120-82   120-82#
T$CODE  41-66*   41-66    41-66    41-66#   41-66#   41-66#   108-32   108-32   108-32   108-32#  108-32#  108-32#  119-64   119-64
       119-64   119-64#  119-64#  119-64#  119-68   119-68   119-68   119-68#  119-68#  119-68#  120-14   120-14   120-14   120-14#
       120-14#  120-14#  120-23   120-23   120-23   120-23#  120-23#  120-23#  120-29   120-29   120-29   120-29#  120-29#  120- 9#
       120-32   120-32   120-32   120-32#  120-32#  120-32#  120-33   120-33   120-33   120-33#  120-33#  120-33#  120-36   120-56
       120-36   120-36#  120-36#  120-36#  120-41   120-41   120-41   120-41#  120-41#  120-41#  120-50   120-50   120-50   120-50#
       120-50#  120-50#  120-51   120-51   120-51   120-51#  120-51#  120-51#  120-54   120-54   120-54   120-54#  120-54#  120-54#
       120-57   120-57   120-57   120-57#  120-57#  120-57#  120-59   120-59   120-59   120-59#  120-59#  120-59#  120-83   120-83
       120-83   120-83#  120-83#  120-83#  120-121  120-121  120-121  120-121#  120-121#  120-121#  120-127  120-127  120-127  120-127#
       120-127#  120-127#  120-131  120-131  120-131  120-131#  120-131#  120-131#  120-146  120-146  120-146  120-146#  120-146#  120-146#
       120-150  120-150  120-150  120-150#  120-150#  120-150#  120-156  120-156  120-156  120-156#  120-156#  120-156#  120-160  120-160
       120-160  120-160#  120-160#  120-160#  120-166  120-166  120-166  120-166#  120-166#  120-166#  120-180  120-180  120-180  120-180#
       120-180#  120-180#  120-184  120-184  120-184  120-184#  120-184#  120-184#  122-1  122-1  122-1  122-1#  122-1#  122-1#
       122-2    122-2    122-2    122-2#   122-2#   122-2#   122-3    122-3    122-3    122-3#   122-3#   122-3#   122-4    122-4
       122-4    122-4#   122-4#   122-4#   122-5    122-5    122-5    122-5#   122-5#   122-5#   122-6    122-6    122-6    122-6#
       122-6#   122-6#   124-1    124-1    124-1    124-1#   124-1#   124-1#   124-2    124-2#   124-3    124-3    124-3    124-3#
       124-3#   124-3#   124-4    124-4    124-4    124-4#   124-4#   124-4#   124-5    124-5    124-5    124-5#   124-5#   124-5#
       124-6    124-6    124-6    124-6#   124-6#   124-6#   124-7    124-7    124-7    124-7#   124-7#   124-7#
T$ERRN   5-16#    28-7     28-7#    32-20    32-20#   34-22    34-22#   34-40    34-40#   35-11    35-11#   35-18    35-18#   35-25
        35-25#   36-6     36-6#    59-24    59-24#   71-3     71-3#    75-36    75-36#   75-44    75-44#   85-18    85-18#   85-37
        85-37#   86-30    86-30#   87-44    87-44#   88-36    88-36#   89-5     89-5#    94-1     94-1#    94-3     94-3#    101-122
       101-122#  109-2    109-2#   109-6    109-6#   109-11   109-11#  112-37   112-37#  114-17   114-17#  114-27   114-27#  115-19
       115-19#   116-41   116-41#  116-61   116-61#  116-67   116-67#
T$EXCP  41-66    41-66#   119-64   119-64#  119-68   119-68#  120-14   120-14#  120-23   120-23#  120-41   120-41#  120-83   120-83#
       120-121  120-121#  120-127  120-127#  120-131  120-131#  120-146  120-146#  120-150  120-150#  120-156  120-156#  120-160  120-160#
       120-180  120-180#  120-184  120-184#  122-1    122-1#   122-2    122-2#   122-3    122-3#   122-4    122-4#   122-5    122-5#
       124-3    124-3#   124-4    124-4#
T$FLAG  99-118   99-118#  99-118#  108-45   108-45   108-45#  108-45#  119-117  119-117  119-117#  119-117#
T$FREE  125-14   126-22#
T$GMAN   5-16#   41-66    41-66#   41-66#   119-64#  119-64#  119-68#  119-68#  120-14#  120-14#  120-23   120-23#  120-23#  120-41#
       120-41#  120-83#  120-83#  120-121#  120-121#  120-127  120-127#  120-127#  120-131  120-131#  120-131#  120-146#  120-146#  120-150#
       120-150#  120-156#  120-156#  120-160#  120-160#  120-180  120-180#  120-180#  120-184  120-184#  120-184#
```

CROSS REFERENCE TABLE (CREF V01-05)

T$HILI	41-66	41-66#	119-64	119-64#	119-68	119-68#	120-14	120-14#	120-23	120-23#	120-41	120-41#	120-83	120-83#
	120-121	120-121#	120-127	120-127#	120-131	120-131#	120-146	120-146#	120-150	120-150#	120-156	120-156#	120-160	120-160#
	120-180	120-180#	120-184	120-184#	122-1	122-1#	122-2	122-2#	122-3	122-3#	122-4	122-4#	122-5	122-5#
	124-3	124-3#	124-4	124-4#										
T$LAST	5-16#	125-14#	126-13											
T$LOLI	41-66	41-66#	119-64	119-64#	119-68	119-68#	120-14	120-14#	120-23	120-23#	120-41	120-41#	120-83	120-83#
	120-121	120-121#	120-127	120-127#	120-131	120-131#	120-146	120-146#	120-150	120-150#	120-156	120-156#	120-160	120-160#
	120-180	120-180#	120-184	120-184#	122-1	122-1#	122-2	122-2#	122-3	122-3#	122-4	122-4#	122-5	122-5#
	124-3	124-3#	124-4	124-4#										
T$LSYM	5-16	5-16#	7-27	8-27	27-17	27-22	27-28	27-32	27-37	27-49	27-54	27-59	27-63	27-67
	27-71	27-75	27-79	27-83	67-48	81-14	82-22	95-9	96-3	99-156	109-15	110-18	111-30	112-40
	112-43	112-46	112-50	112-56	112-66	113-6	117-24	118-14	120-189	122-7	124-11			
T$LTNO	125-14#													
T$NEST	5-16#	5-43	5-43	5-43#	7-10	7-10	7-10#	7-27	7-27	7-27	7-27#	8-10	8-10	8-10#
	8-27	8-27	8-27	8-27#	8-29	8-29	8-29	8-29#	9-48	9-48	9-48#	27-14	27-14	27-14#
	27-17	27-17	27-17	27-17#	27-19	27-19	27-19#	27-22	27-22	27-22	27-22#	27-24	27-24	27-24#
	27-28	27-28	27-28	27-28#	27-30	27-30	27-30#	27-32	27-32	27-32	27-32#	27-34	27-34	27-34#
	27-37	27-37	27-37	27-37#	27-39	27-39	27-39#	27-49	27-49	27-49	27-49#	27-51	27-51	27-51#
	27-54	27-54	27-54	27-54#	27-56	27-56	27-56#	27-59	27-59	27-59	27-59#	27-61	27-61	27-61#
	27-63	27-63	27-63	27-63#	27-65	27-65	27-65#	27-67	27-67	27-67	27-67#	27-69	27-69	27-69#
	27-71	27-71	27-71	27-71#	27-73	27-73	27-73#	27-75	27-75	27-75	27-75#	27-77	27-77	27-77#
	27-79	27-79	27-79	27-79#	27-81	27-81	27-81#	27-83	27-83	27-83	27-83#	67-31	67-31	67-31#
	67-48	67-48	67-48	67-48#	81-10	81-10	81-10#	81-14	81-14	81-14	81-14#	82-18	82-18	82-18#
	82-22	82-22	82-22	82-22#	95-5	95-5	95-5#	95-9	95-9	95-9	95-9#	96-1	96-1	96-1#
	96-3	96-3	96-3	96-3#	98-28	98-28	98-28	98-28#	99-38	99-38	99-38#	99-45	99-45	99-45#
	99-156	99-156	99-156	99-156#	100-8	100-8	100-8#	100-14	100-14	100-14	100-14#	101-8	101-8	101-8#
	109-15	109-15	109-15	109-15#	110-10	110-10	110-10#	110-18	110-18	110-18	110-18#	111-8	111-8	111-8#
	111-30	111-30	111-30	111-30#	111-31	111-31	111-31	111-31#	112-16	112-16	112-16#	112-18	112-18	112-18#
	112-28	112-28	112-28#	112-40	112-40	112-40	112-40#	112-41	112-41	112-41#	112-43	112-43	112-43	112-43#
	112-44	112-44	112-44#	112-46	112-46	112-46	112-46#	112-47	112-47	112-47#	112-50	112-50	112-50	112-50#
	112-51	112-51	112-51#	112-56	112-56	112-56	112-56#	112-57	112-57	112-57#	112-66	112-66	112-66	112-66#
	113-6	113-6	113-6	113-6#	117-3	117-3	117-3#	117-24	117-24	117-24	117-24#	118-3	118-3	118-3#
	118-14	118-14	118-14	118-14#	119-3	119-3	119-3#	120-189	120-189	120-189	120-189#	120-190	120-190	120-190
	120-190#	121-39	121-39	121-39#	121-50	121-50	121-50#	122-7	122-7	122-7	122-7#	123-12	123-12	123-12#
	124-11	124-11	124-11	124-11#	125-15	125-15	125-15	125-15#						
T$NS0	5-43#	8-29	9-48#	98-28	99-38#	111-31	112-16#	120-190	121-39#	125-15				
T$NS1	7-10#	7-27	8-10#	8-27	27-14#	27-17	27-19#	27-22	27-24#	27-28	27-30#	27-32	27-34#	27-37
	27-39#	27-49	27-51#	27-54	27-56#	27-59	27-61#	27-63	27-65#	27-67	27-69#	27-71	27-73#	27-75
	27-77#	27-79	27-81#	27-83	67-31#	67-48	81-10#	81-14	82-18#	82-22	95-5#	95-9	96-1#	96-3
	99-45#	99-156	100-8#	100-14	101-8#	109-15	110-10#	110-18	111-8#	111-30	112-18#	113-6	117-3#	117-24
	118-3#	118-14	119-3#	120-189	121-50#	122-7	123-12#	124-11						
T$NS2	112-28#	112-40	112-41#	112-43	112-44#	112-46	112-47#	112-50	112-51#	112-56	112-57#	112-66		
T$PCNT	126-13#	126-14	126-14	126-14#										
T$PTAB	126-14	126-14#												
T$PTHV	5-77	126-22#												
T$PTNU	5-16#	126-14	126-14#	126-22	126-22									
T$SAVL	5-16#													
T$SEGL	5-16#													
T$SIZE	125-14	126-22#												
T$SUBN	5-16#	112-18#	112-28	112-28	112-28#	112-41	112-41	112-41#	112-44	112-44	112-44#	112-47	112-47	112-47#
	112-51	112-51	112-51#	112-57	112-57	112-57#	117-3#	118-3#	119-3#					
T$TAGL	5-16#													
T$TAGN	5-16#	7-10	7-10	7-10#	8-10	8-10	8-10#	27-14	27-14	27-14#	27-19	27-19	27-19#	27-24
	27-24	27-24#	27-30	27-30	27-30#	27-34	27-34	27-34#	27-39	27-39	27-39#	27-51	27-51	27-51#
	27-56	27-56	27-56#	27-61	27-61	27-61#	27-65	27-65	27-65#	27-69	27-69	27-69#	27-73	27-73
	27-73#	27-77	27-77	27-77#	27-81	27-81	27-81#	67-31	67-31	67-31#	81-10	81-10	81-10#	82-18
	82-18	82-18#	95-5	95-5	95-5#	96-1	96-1	96-1#	99-45	99-45	99-45#	100-8	100-8	100-8#

```
        101-8    101-8    101-8#   110-10   110-10   110-10#  111-8    111-8    111-8#   112-18   112-18   112-18#  112-28   112-28
        112-28#  112-41   112-41   112-41#  112-44   112-44   112-44#  112-47   112-47   112-47#  112-51   112-51   112-51#  112-57
        112-57   112-57#  117-3    117-3    117-3#   118-3    118-3    118-3#   119-3    119-3    119-3#   121-50   121-50   121-50#
        123-12   123-12   123-12#  126-13   126-13   126-13#  126-14   126-14   126-14   126-14   126-14#  126-14#
T$TEMP    6-8      6-8      6-8      6-8      6-8      6-8      6-8      6-8      6-8#     6-8#     6-8#     6-8#     6-8#     7-27
          7-27#    8-27     8-27#    8-29     8-29#    27-17    27-17#   27-22    27-22#   27-28    27-28#   27-32    27-32#   27-37
         27-37#   27-49    27-49#   27-54    27-54#   27-59    27-59#   27-63    27-63#   27-67    27-67#   27-71    27-71#   27-75
         27-75#   27-79    27-79#   27-83    27-83#   41-66    41-66    41-66    41-66#   41-66#   41-66#   67-48    67-48#   81-14
         81-14#   82-22    82-22#   95-9     95-9#    96-3     96-3#    98-28    98-28#   99-118   99-118#  99-156   99-156#  100-14
        100-14#  108-32   108-32   108-32   108-32#  108-32#  108-32#  108-45   108-45#  109-15   109-15#  110-18   110-18#  111-30
        111-30#  111-31   111-31#  112-40   112-40#  112-43   112-43#  112-46   112-46#  112-50   112-50#  112-56   112-56#  112-66
        112-66#  113-6    113-6#   117-24   117-24#  118-14   118-14#  119-64   119-64   119-64   119-64#  119-64#  119-64#  119-68
        119-68   119-68   119-68#  119-68#  119-68#  119-117  119-117# 120-14   120-14   120-14   120-14#  120-14#  120-14#  120-23
        120-23   120-23   120-23#  120-23#  120-23#  120-29   120-29   120-29   120-29#  120-29#  120-29#  120-32   120-32   120-32
        120-32#  120-32#  120-32#  120-33   120-33   120-33   120-33#  120-33#  120-33#  120-36   120-36   120-36   120-36#  120-36#
        120-36#  120-41   120-41   120-41   120-41#  120-41#  120-41#  120-50   120-50   120-50   120-50#  120-50#  120-50#  120-51
        120-51   120-51   120-51#  120-51#  120-51#  120-54   120-54   120-54   120-54#  120-54#  120-54#  120-57   120-57   120-57
        120-57#  120-57#  120-57#  120-59   120-59   120-59   120-59#  120-59#  120-59#  120-83   120-83   120-83   120-83#  120-83#
        120-83#  120-121  120-121  120-121  120-121#  120-121#  120-121#  120-127  120-127  120-127  120-127#  120-127#  120-127#  120-131
        120-131  120-131  120-131#  120-131#  120-131#  120-146  120-146  120-146  120-146#  120-146#  120-146#  120-150  120-150  120-150
        120-150#  120-150#  120-150#  120-156  120-156  120-156  120-156#  120-156#  120-156#  120-160  120-160  120-160  120-160#  120-160#
        120-160#  120-166  120-166  120-166  120-166#  120-166#  120-166#  120-180  120-180  120-180  120-180#  120-180#  120-180#  120-184
        120-184  120-184  120-184#  120-184#  120-184#  120-189  120-189#  120-190  120-190#  122-1    122-1    122-1    122-1#   122-1#
        122-1#   122-2    122-2    122-2    122-2#   122-2#   122-2#   122-3    122-3    122-3    122-3#   122-3#   122-3#   122-4
        122-4    122-4    122-4#   122-4#   122-4#   122-5    122-5    122-5    122-5#   122-5#   122-5#   122-6    122-6    122-6
        122-6#   122-6#   122-6#   122-7    122-7#   124-1    124-1    124-1    124-1#   124-1#   124-1#   124-3    124-3    124-3
        124-3#   124-3#   124-3#   124-4    124-4    124-4    124-4#   124-4#   124-4#   124-5    124-5    124-5    124-5#   124-5#
        124-5#   124-6    124-6    124-6    124-6#   124-6#   124-6#   124-7    124-7    124-7    124-7#   124-7#   124-7#   124-11
        124-11#  125-15   125-15#
T$TEST    5-16#  112-18   112-18   112-18#  112-28   112-41   112-44   112-47   112-51   112-57   117-3    117-3    117-3#   118-3
        118-3    118-3#   119-3    119-3    119-3#   125-14
T$TSTM    5-16#   27-15    27-16    27-17    27-20    27-21    27-22    27-25    27-26    27-27    27-28    27-31    27-32    27-35
         27-36    27-37    27-40    27-41    27-42    27-45    27-49    27-52    27-53    27-54    27-57    27-58    27-59    27-62
         27-63    27-66    27-67    27-70    27-71    27-74    27-75    27-78    27-79    27-82    27-83    28-7     28-8     32-20
         32-21    34-22    34-40    34-43    34-54    35-11    35-18    35-25    36-6     41-32    41-34    41-42    41-43    41-44
         41-45    41-53    41-59    41-62    41-66    41-113   49-28    49-33    49-34    51-27    51-60    51-63    52-41    52-44
         53-21    55-18    56-6     56-14    56-43    56-56    56-64    58-13    58-20    59-24    61-59    61-69    64-39    64-43
         65-4     65-13    65-30    67-27    67-34    67-37    67-48    68-32    71-3     75-28    75-36    75-44    77-17    77-36
         85-18    85-37    86-13    86-24    86-27    86-30    87-44    88-27    88-36    89-5     92-12    93-21    93-24    93-26
         93-31    93-41    93-76    93-83    93-89    93-99    93-102   94-1     94-2     94-3     94-4     97-10    97-11    98-23
         98-24    99-59    99-70    99-71    99-74    99-75    99-88    99-107   99-108   99-156   101-63   101-66   101-69   101-73
        101-79   101-88   101-91   101-114  101-122  101-123  102-1    102-3    102-13   102-15   102-18   102-24   103-10   106-5
        108-12   108-21   108-30   108-32   108-35   108-42   108-45   109-2    109-3    109-6    109-7    109-11   109-12   109-15
        110-18   111-30   112-28   112-30   112-34   112-37   112-40   112-41   112-43   112-44   112-46   112-47   112-50   112-51
        112-56   112-57   112-66   113-6    114-17   114-27   115-11   115-19   116-16   116-17   116-18   116-33   116-41   116-44
        116-55   116-61   116-63   116-67   116-69   116-72   116-77   117-24   118-14   119-12   119-19   119-24   119-36   119-64
        119-68   119-76   119-113  119-116  119-117  120-12   120-14   120-23   120-29   120-32   120-33   120-36   120-41   120-50
        120-51   120-54   120-57   120-59   120-77   120-78   120-79   120-80   120-81   120-82   120-83   120-121  120-127  120-131
        120-146  120-150  120-156  120-160  120-166  120-180  120-184  120-189
T$TSTS    5-16#  112-18#  117-3#   118-3#   119-3#
T1        6-8    112-18#
T1.1    112-28#
T1.2    112-41#
T1.3    112-44#
T1.4    112-47#
T1.5    112-51#
T1.6    112-57#
```

CROSS REFERENCE TABLE (CREF V01-05)

```
T1ERR   113-5#
T1GOOD  112-36    112-39#
T1MSIZ   36-46     39-26#
T1NAME   24-32#    56-75
T1NEXT  112-22#   113-3
T1SKIP  112-24    112-38    113-1#
T2        6-8     117-3#
T2CMD    36-48     41-29#
T2CMD0   41-33     41-36#
T2CMD2   41-38     41-50#
T2CMD3   41-72     41-84#
T2CMD9   41-34#
T2CMDE   41-79     41-88     41-91     41-93     41-108    41-113#   66-47
T2CMDM   41-30     41-32#
T2CMDN   41-98     41-100#
T2CMDQ   41-51     41-66#    41-114
T2CMDR   41-86     41-95#
T2CMDV   41-70     41-78#
T2CMDW   41-101    41-105    41-107#
T2CMDX   41-31     41-35     41-80     41-111#
T2CMS1   26-61#    41-42
T2CMS2   26-62#    41-43
T2CMS3   26-65#    41-44
T2CMS4   26-67#    41-45
T2CMS5   26-70#    41-113
T2DLL    36-47     40-37#
T2DR     23-16#    41-48*    41-76*    41-77
T2GND1   66-16     66-19#
T2GND2   66-20#    66-41
T2GND3   66-25     66-31#
T2GNE    66-23     66-27     66-29     66-45#
T2GNUM   41-74     41-78     41-90     41-100    41-104    41-107    66-12#    66-18
T2GNX    66-14     66-39     66-42#
T2NAME   24-33#    56-76
T2NEXT  117-10#   117-22
T2PNT    65-7      65-9      65-17#
T2PNTB   41-61     65-12#
T2PNTD   65-27     65-29#
T2PNTO   65-22     65-24#
T2PNTW   41-55     41-57     65-3#
T2WARN   26-60#    41-34
T2WRO    23-15#    41-47*    41-56     41-96     41-110*
T2WRR    23-14#    41-46*    41-54     41-95     41-109*
T3        6-8     118-3#
T3NAME   24-34#    56-77
T4        6-8     119-3#
T4BB     25-14#   120-14
T4BB1    36-51     45-22#
T4BB1E   45-24     45-30#
T4BB2    36-52     46-15#
T4BB2E   46-17     46-23#
T4BBI    25-15#   120-23
T4BE     25-27#   120-121
T4BEG    25-28#   120-127
T4CON   119-20    119-23    119-25#
T4CYL    25-34#   120-166
T4CYLB   25-35#   120-180
```

CROSS REFERENCE TABLE (CREF V01-05)

```
T4CYLE   25-36#  120-184
T4DCA    25-22#  120-51
T4DCR    25-23#  120-54
T4DEF   119-35   119-80#
T4DEFA  119-81#  119-97
T4DEFB  119-84#  119-94
T4DEFC  119-90#  119-92
T4DEFD  119-93#
T4DEFE  119-85   119-95#
T4DEFW  119-37   119-76#
T4DP     25-20#  120-41
T4DPC    25-37#  119-64
T4DPD    25-38#  119-68
T4ECC    25-21#  120-50
T4END    25-29#  120-131
T4EXIT  119-112  119-116#
T4GRC    25-32#  120-156
T4GRP    25-33#  120-160
T4MPRM   36-49    42-18#
T4MXFR   36-55    49-18#
T4NAME   24-35#   56-78
T4OPT1   26-50#  120-77
T4OPT2   26-51#  120-78
T4OPT3   26-52#  120-79
T4OPT4   26-53#  120-80
T4OPT5   26-54#  120-81
T4OPT6   26-55#  120-82
T4OPT7   25-26#   41-66   120-83
T4PRM1  119-43#  119-58
T4PRM2  119-46#  119-55
T4PRM3  119-49   119-52   119-54#
T4PRM4  119-47   119-56#
T4PRM5  119-67#  119-71
T4Q01   120-22#  120-25   120-27
T4Q02   120-16   120-28#
T4Q03   120-32#
T4Q04   120-35   120-38#
T4Q05   120-37   120-39#
T4Q06   120-31   120-44#
T4Q07   120-45   120-47   120-50#
T4Q08   120-53   120-56#
T4Q09   120-49   120-55   120-57#
T4Q10   120-65   120-68#
T4Q11   120-67   120-69   120-72   120-75   120-77#
T4Q12   120-85   120-88#
T4Q13   120-89   120-95#
T4Q14   120-98   120-102#
T4Q15   120-105  120-107#
T4Q16   120-94   120-108  120-111#
T4Q17   120-112#
T4Q18   120-115# 120-117
T4Q19   120-87   120-91   120-101  120-110  120-118#
T4Q20   120-126# 120-129  120-135
T4Q21   120-130# 120-133
T4Q22   120-119  120-137#
T4Q23   120-149# 120-153
T4Q24   120-144  120-155#
```

CROSS REFERENCE TABLE (CREF V01-05)

```
T4Q25   120-159# 120-163
T4Q26   120-154  120-164#
T4Q27   120-140  120-168  120-174#
T4Q27A  120-175  120-177#
T4Q28   120-179# 120-182
T4Q29   120-183# 120-186
T4Q30   120-136  120-138  120-173  120-187#
T4QHED   26-49#  120-12
T4QUES  119-50   120-11#
T4RET    25-24#  120-57
T4RO     25-16#  120-29
T4RUN   119-33   119-63   119-72   119-101#
T4SEEK   36-54    48-1#
T4SEK    25-25#  120-59
T4SOFT   36-53    47-13#
T4STRT  119-6    119-8    119-26   119-32#
T4TRAK   25-31#  120-150
T4TRC    25-30#  120-146
T4UPRE   44-4     44-82#
T4UPRM   36-50    44-1#
T4UPRX   44-33    44-54    44-66    44-76#
T4WAIT  119-28   119-109  119-111# 119-114
T4WARN   26-48#  119-76
T4WCA    25-18#  120-33
T4WCR    25-19#  120-36
T4WO     25-17#  120-32
TEMP     22-58#   41-66    41-67    63-20    64-18    99-93    99-107  108-32   108-33   119-67*  119-68   119-69   120-13*  120-14
        120-15   120-20   120-23   120-28*  120-29   120-30   120-32   120-33   120-34   120-36   120-38*  120-39   120-40*  120-41
        120-42   120-43*  120-44   120-46   120-48   120-50   120-51   120-52   120-54   120-56*  120-57   120-58*  120-59   120-60*
        120-61   120-63*  120-66*  120-70*  120-73*  120-76*  120-83   120-84*  120-88*  120-97   120-104* 120-120* 120-121  120-122
        120-123  120-127  120-131  120-145* 120-146  120-147  120-149* 120-150  120-151  120-155* 120-156  120-157  120-159* 120-160
        120-161  120-164* 120-165* 120-166  120-167  120-180  120-184
TINIT    31-14#  112-20   117-6    118-6    119-104
TINITR   31-19    31-24#
TNAMES   56-75#   67-34
TNUM     22-44#   31-18    31-26*   51-36    67-32    68-40    99-59    99-72    102-5*   119-5
TOOMEM   25-12#  109-11
TOOMER  107-18   109-11#
TSTTAB   22-31#   33-21    34-6     112-21*  112-22   113-1*   113-2    117-8*   117-20*  117-21   118-8*   119-106*
UAM       9-70#
UCNT     22-47#   33-20*   33-30*   34-7*    34-62*   119-41*  119-53*  119-62
UDAI1    84-39    85-3#
UDAI1A   85-15#   85-21
UDAI1B   85-17    85-20#
UDAI1C   85-16    85-22#
UDAI1L   85-5#    85-7
UDAI2    85-29#
UDAI2E   85-32    85-37#
UDAI2L   85-31#   85-34
UDAI3    85-35    85-42#
UDAID1   86-16*   87-4#    87-20    87-29
UDAID2   86-17*   86-18*   87-6#
UDAID3   87-8#
UDAIDT   86-37    87-3#
UDAIEX   85-12    85-19    85-25    85-38    85-52#
UDAINT   68-36    84-23#   112-49   112-53
UDAIR1   87-3     87-14#
```

CROSS REFERENCE TABLE (CREF V01-05)

```
UDAIR2  87-5     87-20#
UDAIR3  87-7     87-29#
UDAIR4  87-9     87-37#
UDAIRC  87-15    87-24    87-32    87-42#
UDAIRX  87-43    87-46#
UDAISE  86-32    86-42    86-44    86-52#
UDAISG  86-29    86-36#
UDAISL  86-41#   86-49
UDAIST  85-11    86-13#
UDAISX  86-47    86-51#
UDARS1  88-25#   88-29    88-32    88-34
UDARS2  88-26    89-3#
UDARSD  86-36*   86-45*   86-46    88-19*   88-25    89-16#  114-8*   116-24*
UDARSE  88-37    89-6#
UDARSP  85-24    86-41    88-18#  114-9    116-25
UDARSX  89-4     89-11#
UDASRV  82-18#  104-14
UF.576  15-12#
UF.CMR  15-3#
UF.CMW  15-4#
UF.INA  15-6#
UF.RPL  15-5#
UF.SCH  15-7#
UF.SCL  15-8#
UF.WBN  15-9#
UF.WPH  15-10#
UF.WPS  15-11#
URNING  22-46#   33-16*   33-28*   33-35    34-69*  119-25
URUN    22-45#   33-15*   33-20    34-7
UTOT1   50-19#   50-22    50-26
UTOT2   50-20    50-27#
UTOT3   50-32#   50-34
UTOT4   50-29    50-35#
UTOTST  36-56    50-16#
WAITMS  68-54    70-24    75-1 #
WCHNG  114-13   114-23   115-5#
WCHNGD 114-11*  114-21*  115-9    115-24#
X$ALWA   5-16#
X$FALS   5-16#
X$OFFS   5-16#
X$TRUE   5-16#
```

CROSS REFERENCE TABLE (CREF V01-05)

CROSS REFERENCE TABLE (CREF V01-05)

```
        115-19  116-41  116-61  116-67
ERRHRD   35-11   35-18   35-25   36-6    71-3
ERROR    67-27
ERRSF    28-7    32-20   94-1    94-3    101-122 109-2   109-6   109-11
ERRTBL   22-21
EXIT     99-118  108-45  119-117
GETBYT   93-26   93-31   93-41   93-76   93-83   93-89   93-99   93-102
GMANID   41-66   119-64  119-68  120-14  120-23  120-41  120-83  120-121 120-127 120-131 120-146 120-150 120-156 120-160
        120-180 120-184
GMANIL  108-32  120-29  120-32  120-33  120-36  120-50  120-51  120-54  120-57  120-59  120-166
GPHARD  101-114 103-10  106-5
GPRMA   122-1   122-2
GPRMD    41-66   41-66#  119-64  119-64# 119-68  119-68# 120-14  120-14# 120-23  120-23# 120-41  120-41# 120-83  120-83#
        120-121 120-121# 120-127 120-127# 120-131 120-131# 120-146 120-146# 120-150 120-150# 120-156 120-156# 120-160 120-160#
        120-180 120-180# 120-184 120-184# 122-3   122-4   122-5   124-3   124-4
GPRML   108-32  108-32#  120-29  120-29#  120-32  120-32#  120-33  120-33#  120-36  120-36#  120-50  120-50#  120-51  120-51#
        120-54  120-54#  120-57  120-57#  120-59  120-59#  120-166 120-166# 122-6   124-1   124-5   124-6   124-7
HEADER   5-77
LASTAD  125-14
M$BYTE   5-77    5-77    5-77    5-77#
M$CHEC   99-118  99-118#  108-45  108-45#  119-117 119-117#
M$CNTO   41-66   41-66#   108-32  108-32#  119-64  119-64#  119-68  119-68#  120-14  120-14#  120-23  120-23#  120-29  120-29#
        120-32  120-32#  120-33  120-33#  120-36  120-36#  120-41  120-41#  120-50  120-50#  120-51  120-51#  120-54  120-54#
        120-57  120-57#  120-59  120-59#  120-83  120-83#  120-121 120-121# 120-127 120-127# 120-131 120-131# 120-146 120-146#
        120-150 120-150# 120-156 120-156# 120-160 120-160# 120-166 120-166# 120-180 120-180# 120-184 120-184# 122-1   122-1#
        122-2   122-2#   122-3   122-3#   122-4   122-4#   122-5   122-5#   122-6   122-6#   124-1   124-1#   124-3   124-3#
        124-4   124-4#   124-5   124-5#   124-6   124-6#   124-7   124-7#
M$COUN   27-15   27-15#   27-16   27-16#   27-20   27-20#   27-21   27-21#   27-25   27-25#   27-26   27-26#   27-27   27-27#
        27-31   27-31#   27-35   27-35#   27-36   27-36#   27-40   27-40#   27-41   27-41#   27-42   27-42#   27-45   27-45
        27-45#  27-52    27-52#  27-53    27-53#  27-57    27-57#  27-58    27-58#  27-62    27-62    27-62#  27-66   27-66
        27-66#  27-70    27-70#  27-74    27-74#  27-78    27-78#  27-82    27-82   27-82#   41-34    41-34#  41-42   41-42
        41-42   41-42#   41-43   41-43#   41-44   41-44#   41-45   41-45#   41-53   41-53#   41-59    41-59#  41-62   41-62#
        41-113  41-113#  49-33   49-33    49-33   49-33#   49-34   49-34#   51-60   51-60#   51-63    51-63#  52-44   52-44#
        53-21   53-21    53-21   53-21#   55-18   55-18#   56-6    56-6#    56-14   56-14#   56-43    56-43#  56-56   56-56#
        56-64   56-64#   58-13   58-13#   58-20   58-20#   61-59   61-59#   61-69   61-69#   64-39    64-39#  64-43   64-43#
        65-4    65-4#    65-13   65-13#   65-30   65-30#   67-34   67-34    67-34   67-34#   67-37    67-37#  98-23   98-23
        98-23   98-23#   98-24   98-24#   99-59   99-59#   99-70   99-70    99-70   99-70#   99-71    99-71#  99-74   99-74#
        99-75   99-75#   99-88   99-88    99-88#  99-107   99-107  99-107   99-107  99-107#  99-108   99-108  99-108  99-108#
        102-18  102-18#  108-12  108-12#  108-21  108-21   108-21  108-21#  116-16  116-16   116-16#  116-69  116-69#  119-12
        119-12# 119-19   119-19# 119-24   119-24# 119-76   119-76# 120-12   120-12   120-12   120-12#  120-77  120-77#  120-78
        120-78# 120-79   120-79# 120-80   120-80# 120-81   120-81# 120-82   120-82#
M$DATA   5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77
         5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77
         5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77    5-77#   5-77#   24-12   24-12#  24-22
        24-22#
M$DECR   7-27    7-27#   8-27    8-27#   8-29    8-29#   27-17   27-17#  27-22   27-22#  27-28   27-28#  27-32   27-32#
        27-37   27-37#   27-49   27-49#   27-54   27-54#   27-59   27-59#   27-63   27-63#   27-67   27-67#   27-71   27-71#
        27-75   27-75#   27-79   27-79#   27-83   27-83#   67-48   67-48#   81-14   81-14#   82-22   82-22#   95-9    95-9#
        96-3    96-3#    98-28   98-28#   99-156  99-156#  100-14  100-14#  109-15  109-15#  110-18  110-18#  111-30  111-30#
        111-31  111-31#  112-40  112-40#  112-43  112-43#  112-46  112-46#  112-50  112-50#  112-56  112-56#  112-66  112-66#
        113-6   113-6#   117-24  117-24#  118-14  118-14#  120-189 120-189# 120-190 120-190# 122-7   122-7#   124-11  124-11#
        125-15  125-15#  126-14  126-14#
M$DEFA   41-66   41-66#   108-32  108-32#  119-64  119-64#  119-68  119-68#  120-14  120-14#  120-23  120-23#  120-29  120-29#
        120-32  120-32#  120-33  120-33#  120-36  120-36#  120-41  120-41#  120-50  120-50#  120-51  120-51#  120-54  120-54#
        120-57  120-57#  120-59  120-59#  120-83  120-83#  120-121 120-121# 120-127 120-127# 120-131 120-131# 120-146 120-146#
        120-150 120-150# 120-156 120-156# 120-160 120-160# 120-166 120-166# 120-180 120-180# 120-184 120-184# 122-1   122-1#
        122-2   122-2#   122-3   122-3#   122-4   122-4#   122-5   122-5#   122-6   122-6#   124-1   124-1#   124-3   124-3#
```

CROSS REFERENCE TABLE (CREF V01-05)

	124-4	124-4#	124-5	124-5#	124-6	124-6#	124-7	124-7#						
M$ENDE	7-27#	8-27#	8-29#	27-17#	27-22#	27-28#	27-32#	27-37#	27-49#	27-54#	27-59#	27-63#	27-67#	27-71#
	27-75#	27-79#	27-83#	67-48#	81-14#	82-22#	95-9#	96-3#	98-28#	99-156#	109-15#	110-18#	111-30#	111-31#
	112-40#	112-43#	112-46#	112-50#	112-56#	112-66#	113-6#	117-24#	118-14#	120-189#	120-190#	122-7#	124-11#	125-15#
M$ERRI	28-7	28-7#	32-20	32-20#	34-22	34-22#	34-40	34-40#	35-11	35-11#	35-18	35-18#	35-25	35-25#
	36-6	36-6#	59-24	59-24#	71-3	71-3#	75-36	75-36#	75-44	75-44#	85-18	85-18#	85-37	85-37#
	86-30	86-30#	87-44	87-44#	88-36	88-36#	89-5	89-5#	94-1	94-1#	94-3	94-3#	101-122	101-122#
	109-2	109-2#	109-6	109-6#	109-11	109-11#	112-37	112-37#	114-17	114-17#	114-27	114-27#	115-19	115-19#
	116-41	116-41#	116-61	116-61#	116-67	116-67#								
M$EXCP	41-66	41-66	41-66#	119-64	119-64	119-64#	119-68	119-68	119-68#	120-14	120-14	120-14#	120-23	120-23
	120-23#	120-41	120-41	120-41#	120-83	120-83	120-83#	120-121	120-121	120-121#	120-127	120-127	120-127#	120-131
	120-131	120-131#	120-146	120-146	120-146#	120-150	120-150	120-150#	120-156	120-156	120-156#	120-160	120-160	120-160#
	120-180	120-180	120-180#	120-184	120-184	120-184#	122-1	122-1	122-1#	122-2	122-2	122-2#	122-3	122-3
	122-3#	122-4	122-4	122-4#	122-5	122-5	122-5#	124-3	124-3	124-3#	124-4	124-4	124-4#	
M$EXIT	99-118#	108-45	108-45#	119-117	119-117#									
M$EXSE	99-118#	108-45#	119-117#											
M$EXTJ	99-118	99-118#	108-45#	119-117#										
M$GEN	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77
	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77
	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77#	5-77#	5-77#	5-77#
	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#
	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#
	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	6-8	6-8#	7-10	7-10	7-10#	7-10#	7-27	7-27#
	8-10	8-10	8-10#	8-10#	8-27	8-27#	22-21	22-21#	24-12	24-12#	24-22	24-22#	27-14	27-14#
	27-17	27-17#	27-19	27-19#	27-22	27-22#	27-24	27-24#	27-28	27-28#	27-30	27-30#	27-32	27-32#
	27-34	27-34#	27-37	27-37#	27-39	27-39#	27-49	27-49#	27-51	27-51#	27-54	27-54#	27-56	27-56#
	27-59	27-59#	27-61	27-61#	27-63	27-63#	27-65	27-65#	27-67	27-67#	27-69	27-69#	27-71	27-71#
	27-73	27-73#	27-75	27-75#	27-77	27-77#	27-79	27-79#	27-81	27-81#	27-83	27-83#	41-66	41-66#
	67-31	67-31#	67-48	67-48#	81-10	81-10#	81-14	81-14#	82-18	82-18#	82-22	82-22#	95-5	95-5#
	95-9	95-9#	96-1	96-1#	96-3	96-3#	99-45	99-45#	99-156	99-156#	100-8	100-8#	101-8	101-8#
	108-32	108-32#	109-15	109-15#	110-10	110-10#	110-18	110-18#	111-8	111-8#	111-30	111-30#	112-18	112-18#
	112-28	112-28#	112-40	112-40#	112-41	112-41#	112-43	112-43#	112-44	112-44#	112-46	112-46#	112-47	112-47#
	112-50	112-50#	112-51	112-51#	112-56	112-56#	112-57	112-57#	112-66	112-66#	113-6	113-6#	117-3	117-3#
	117-24	117-24#	118-3	118-3#	118-14	118-14#	119-3	119-3#	119-64	119-64#	119-68	119-68#	120-14	120-14#
	120-23	120-23#	120-29	120-29#	120-32	120-32#	120-33	120-33#	120-36	120-36#	120-41	120-41#	120-50	120-50#
	120-51	120-51#	120-54	120-54#	120-57	120-57#	120-59	120-59#	120-83	120-83#	120-121	120-121#	120-127	120-127#
	120-131	120-131#	120-146	120-146#	120-150	120-150#	120-156	120-156#	120-160	120-160#	120-166	120-166#	120-180	120-180#
	120-184	120-184#	120-189	120-189#	121-50	121-50#	122-7	122-7#	123-12	123-12#	124-11	124-11#	125-14	125-14#
	126-14	126-14#	126-21	126-21#										
M$GENB	41-66	41-66#	108-32	108-32#	119-64	119-64#	119-68	119-68#	120-14	120-14#	120-23	120-23#	120-29	120-29#
	120-32	120-32#	120-33	120-33#	120-36	120-36#	120-41	120-41#	120-50	120-50#	120-51	120-51#	120-54	120-54#
	120-57	120-57#	120-59	120-59#	120-83	120-83#	120-121	120-121#	120-127	120-127#	120-131	120-131#	120-146	120-146#
	120-150	120-150#	120-156	120-156#	120-160	120-160#	120-166	120-166#	120-180	120-180#	120-184	120-184#		
M$GETS	7-27	7-27#	8-27	8-27#	8-29	8-29#	27-17	27-17#	27-22	27-22#	27-28	27-28#	27-32	27-32#
	27-37	27-37#	27-49	27-49#	27-54	27-54#	27-59	27-59#	27-63	27-63#	27-67	27-67#	27-71	27-71#
	27-75	27-75#	27-79	27-79#	27-83	27-83#	67-48	67-48#	81-14	81-14#	82-22	82-22#	95-9	95-9#
	96-3	96-3#	98-28	98-28#	99-156	99-156#	100-14	100-14#	109-15	109-15#	110-18	110-18#	111-30	111-30#
	111-31	111-31#	112-40	112-40#	112-43	112-43#	112-46	112-46#	112-50	112-50#	112-56	112-56#	112-66	112-66#
	113-6	113-6#	117-24	117-24#	118-14	118-14#	120-189	120-189#	120-190	120-190#	122-7	122-7#	124-11	124-11#
	125-15	125-15#												
M$GETT	99-118#	108-45#	119-117#											
M$GNGB	5-43#	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77
	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77
	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77#	5-77#	5-77#
	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#
	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#
	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	6-8	6-8#	7-10	7-10	7-10#	8-10	8-10

CROSS REFERENCE TABLE (CREF V01-05)

	8-10#	9-48#	22-21	22-21#	24-12	24-12#	24-22	24-22#	27-14	27-14#	27-19	27-19#	27-24	27-24#
	27-30	27-30#	27-34	27-34#	27-39	27-39#	27-51	27-51#	27-56	27-56#	27-61	27-61#	27-65	27-65#
	27-69	27-69#	27-73	27-73#	27-77	27-77#	27-81	27-81#	67-31	67-31#	81-10	81-10#	82-18	82-18#
	95-5	95-5#	96-1	96-1#	99-38#	99-45	99-45#	100-8	100-8#	101-8	101-8#	110-10	110-10#	111-8
	111-8#	112-16#	121-39#	121-50	121-50#	123-12	123-12#	125-14	125-14#					
M$GNIN	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77
	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77
	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77	5-77
	5-77	5-77	5-77	5-77	5-77	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#
	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#
	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#
	5-77#	5-77#	5-77#	5-77#	5-77#	5-77#	6-8	6-8	6-8	6-8	6-8	6-8#	6-8#	6-8#
	6-8#	6-8#	7-10	7-10#	8-10	8-10#	24-12	24-12	24-12#	24-12#	24-22	24-22	24-22#	24-22#
	27-15	27-15	27-15	27-15	27-15	27-15#	27-15#	27-15#	27-15#	27-16	27-16	27-16	27-16	27-16
	27-16	27-16#	27-16#	27-16#	27-16#	27-16#	27-17	27-17#	27-20	27-20	27-20	27-20	27-20	27-20#
	27-20#	27-20#	27-20#	27-21	27-21	27-21	27-21	27-21	27-21	27-21#	27-21#	27-21#	27-21#	27-21#
	27-22	27-22#	27-25	27-25	27-25	27-25	27-25	27-25#	27-25#	27-25#	27-25#	27-26	27-26	27-26
	27-26	27-26	27-26	27-26#	27-26#	27-26#	27-26#	27-26#	27-27	27-27	27-27	27-27	27-27	27-27
	27-27#	27-27#	27-27#	27-27#	27-27#	27-28	27-28#	27-31	27-31	27-31	27-31	27-31	27-31	27-31#
	27-31#	27-31#	27-31#	27-31#	27-32	27-32#	27-35	27-35	27-35	27-35	27-35	27-35#	27-35#	27-35#
	27-35#	27-36	27-36	27-36	27-36	27-36	27-36	27-36#	27-36#	27-36#	27-36#	27-36#	27-37	27-37#
	27-40	27-40	27-40	27-40	27-40	27-40#	27-40#	27-40#	27-40#	27-41	27-41	27-41	27-41	27-41
	27-41#	27-41#	27-41#	27-41#	27-42	27-42	27-42	27-42	27-42	27-42#	27-42#	27-42#	27-42#	27-45
	27-45	27-45	27-45	27-45	27-45	27-45	27-45#	27-45#	27-45#	27-45#	27-45#	27-45#	27-49	27-49#
	27-52	27-52	27-52	27-52	27-52	27-52	27-52#	27-52#	27-52#	27-52#	27-52#	27-53	27-53	27-53
	27-53	27-53	27-53#	27-53#	27-53#	27-53#	27-54	27-54#	27-57	27-57	27-57	27-57	27-57	27-57
	27-57#	27-57#	27-57#	27-57#	27-57#	27-58	27-58	27-58	27-58	27-58	27-58#	27-58#	27-58#	27-58#
	27-59	27-59#	27-62	27-62	27-62	27-62	27-62	27-62	27-62	27-62#	27-62#	27-62#	27-62#	27-62#
	27-62#	27-63	27-63#	27-66	27-66	27-66	27-66	27-66	27-66	27-66	27-66#	27-66#	27-66#	27-66#
	27-66#	27-66#	27-67	27-67#	27-70	27-70	27-70	27-70	27-70	27-70	27-70#	27-70#	27-70#	27-70#
	27-70#	27-71	27-71#	27-74	27-74	27-74	27-74	27-74	27-74	27-74#	27-74#	27-74#	27-74#	27-74#
	27-75	27-75#	27-78	27-78	27-78	27-78	27-78	27-78	27-78#	27-78#	27-78#	27-78#	27-78#	27-79
	27-79#	27-82	27-82	27-82	27-82	27-82	27-82	27-82	27-82#	27-82#	27-82#	27-82#	27-82#	27-82#
	27-83	27-83#	28-7	28-7	28-7	28-7	28-7#	28-7#	28-7#	28-7#	28-7#	28-8	28-8#	32-20
	32-20	32-20	32-20	32-20#	32-20#	32-20#	32-20#	32-20#	32-21	32-21#	34-22	34-22	34-22	34-22
	34-22#	34-22#	34-22#	34-22#	34-22#	34-40	34-40	34-40	34-40	34-40#	34-40#	34-40#	34-40#	34-40#
	34-43	34-43#	34-54	34-54#	35-11	35-11	35-11	35-11	35-11#	35-11#	35-11#	35-11#	35-11#	35-18
	35-18	35-18	35-18	35-18#	35-18#	35-18#	35-18#	35-18#	35-25	35-25	35-25	35-25	35-25#	35-25#
	35-25#	35-25#	35-25#	36-6	36-6	36-6	36-6	36-6#	36-6#	36-6#	36-6#	36-6#	41-32	41-32#
	41-33	41-33#	41-34	41-34	41-34	41-34	41-34	41-34#	41-34#	41-34#	41-34#	41-42	41-42	41-42
	41-42	41-42	41-42	41-42	41-42	41-42#	41-42#	41-42#	41-42#	41-42#	41-42#	41-42#	41-43	41-43
	41-43	41-43	41-43	41-43#	41-43#	41-43#	41-43#	41-44	41-44	41-44	41-44	41-44	41-44#	41-44#
	41-44#	41-44#	41-45	41-45	41-45	41-45	41-45	41-45#	41-45#	41-45#	41-45#	41-53	41-53	41-53
	41-53	41-53	41-53	41-53#	41-53#	41-53#	41-53#	41-53#	41-59	41-59	41-59	41-59	41-59	41-59
	41-59#	41-59#	41-59#	41-59#	41-59#	41-62	41-62	41-62	4 -62	41-62	41-62#	41-62#	41-62#	41-62#
	41-66	41-66	41-66	41-66	41-66	41-66	41-66	41-66	41-66#	41-66#	41-66#	41-66#	41-113	41-113
	41-113	41-113	41-113	41-113#	41-113#	41-113#	41-113#	49-28	49-28#	49-28#	49-33	49-33	49-33	49-33
	49-33	49-33	49-33	49-33	49-33#	49-33#	49-33#	49-33#	49-33#	49-33#	49-33#	49-34	49-34	49-34
	49-34	49-34	49-34#	49-34#	49-34#	49-34#	51-27	51-27#	51-27#	51-60	51-60	51-60	51-60	51-60
	51-60#	51-60#	51-60#	51-60#	51-63	51-63	51-63	51-63	51-63	51-63#	51-63#	51-63#	51-63#	52-41
	52-41#	52-41#	52-44	52-44	52-44	52-44	52-44	52-44	52-44#	52-44#	52-44#	52-44#	52-44#	53-21
	53-21	53-21	53-21	53-21	53-21	53-21	53-21	53-21#	53-21#	53-21#	53-21#	53-21#	53-21#	53-21#
	55-18	55-18	55-18	55-18	55-18	55-18#	55-18#	55-18#	55-18#	56-6	56-6	56-6	56-6	56-6
	56-6	56-6#	56-6#	56-6#	56-6#	56-6#	56-14	56-14	56-14	56-14	56-14	56-14	56-14#	56-14#
	56-14#	56-14#	56-14#	56-43	56-43	56-43	56-43	56-43	56-43#	56-43#	56-43#	56-43#	56-56	56-56
	56-56	56-56	56-56	56-56#	56-56#	56-56#	56-56#	56-64	56-64	56-64	56-64	56-64	56-64	56-64#
	56-64#	56-64#	56-64#	56-64#	58-13	58-13	58-13	58-13	58-13	58-13#	58-13#	58-13#	58-13#	58-20

CROSS REFERENCE TABLE (CREF V01-05)

58-20 58-20 58-20 58-20 58-20# 58-20# 58-20# 58-20# 59-24 59-24 59-24 59-24 59-24# 59-24#
59-24# 59-24# 59-24# 61-59 61-59 61-59 61-59 61-59 61-59 61-59# 61-59# 61-59# 61-59# 61-59#
61-69 61-69 61-69 61-69 61-69 61-69 61-69# 61-69# 61-69# 61-69# 61-69# 64-39 64-39 64-39
64-39 64-39 64-39# 64-39# 64-39# 64-39# 64-43 64-43 64-43 64-43 64-43 64-43# 64-43# 64-43#
64-43# 65-4 65-4 65-4 65-4 65-4 65-4 65-4# 65-4# 65-4# 65-4# 65-4# 65-13 65-13
65-13 65-13 65-13 65-13 65-13# 65-13# 65-13# 65-13# 65-13# 65-30 65-30 65-30 65-30 65-30
65-30 65-30# 65-30# 65-30# 65-30# 65-30# 67-27 67-27# 67-34 67-34 67-34 67-34 67-34 67-34
67-34 67-34 67-34# 67-34# 67-34# 67-34# 67-34# 67-34# 67-37 67-37 67-37 67-37 67-37 67-37
67-37 67-37# 67-37# 67-37# 67-37# 67-37# 67-48 67-48# 68-32 68-32 68-32 68-32 68-32 68-32
68-32# 68-32# 68-32# 68-32# 68-32# 68-32# 71-3 71-3 71-3 71-3 71-3# 71-3# 71-3# 71-3#
71-3# 75-28 75-28# 75-36 75-36 75-36 75-36 75-36# 75-36# 75-36# 75-36# 75-36# 75-44 75-44
75-44 75-44 75-44# 75-44# 75-44# 75-44# 75-44# 77-17 77-17 77-17 77-17 77-17 77-17 77-17#
77-17# 77-17# 77-17# 77-17# 77-17# 77-36 77-36 77-36# 77-36# 81-14 81-14# 82-22 82-22# 85-18
85-18 85-18 85-18 85-18# 85-18# 85-18# 85-18# 85-18# 85-37 85-37 85-37 85-37 85-37# 85-37#
85-37# 85-37# 85-37# 86-13 86-13# 86-24 86-24 86-24 86-24 86-24 86-24# 86-24# 86-24#
86-24# 86-24# 86-24# 86-27 86-27 86-27# 86-27# 86-30 86-30 86-30 86-30 86-30# 86-30# 86-30#
86-30# 87-44 87-44 87-44 87-44 87-44# 87-44# 87-44# 87-44# 87-44# 88-27 88-27# 88-36
88-36 88-36 88-36 88-36# 88-36# 88-36# 88-36# 88-36# 89-5 89-5 89-5 89-5# 89-5#
89-5# 89-5# 89-5# 92-12 92-12# 93-21 93-21 93-21# 93-21# 93-24 93-24# 93-26 93-26 93-26#
93-26# 93-31 93-31# 93-31# 93-41 93-41# 93-41# 93-76 93-76# 93-76# 93-83 93-83# 93-83# 93-89
93-89# 93-89# 93-99 93-99# 93-99# 93-102 93-102# 93-102# 94-1 94-1 94-1 94-1 94-1# 94-1#
94-1# 94-1# 94-1# 94-2 94-2# 94-3 94-3 94-3 94-3# 94-3# 94-3# 94-3# 94-3#
94-4 94-4# 95-9 95-9# 96-3 96-3# 97-10 97-10# 97-11 97-11# 98-23 98-23 98-23 98-23
98-23 98-23 98-23 98-23 98-23# 98-23# 98-23# 98-23# 98-23# 98-23# 98-24 98-24 98-24
98-24 98-24 98-24# 98-24# 98-24# 98-24# 99-59 99-59 99-59 99-59 99-59 99-59 99-59# 99-59#
99-59# 99-59# 99-59# 99-70 99-70 99-70 99-70 99-70 99-70 99-70 99-70# 99-70# 99-70#
99-70# 99-70# 99-70# 99-70# 99-71 99-71 99-71 99-71 99-71 99-71# 99-71# 99-71# 99-74
99-74 99-74 99-74 99-74 99-74# 99-74# 99-74# 99-74# 99-75 99-75 99-75 99-75 99-75 99-75#
99-75# 99-75# 99-75# 99-88 99-88 99-88 99-88 99-88 99-88 99-88 99-88# 99-88# 99-88# 99-88#
99-88# 99-88# 99-107 99-107 99-107 99-107 99-107 99-107 99-107 99-107 99-107 99-107# 99-107# 99-107#
99-107# 99-107# 99-107# 99-107# 99-107# 99-108 99-108 99-108 99-108 99-108 99-108 99-108 99-108#
99-108# 99-108# 99-108# 99-108# 99-108# 99-108# 99-118 99-118 99-118# 99-118# 99-156 99-156# 101-63 101-63
101-63# 101-63# 101-64 101-64# 101-66 101-66 101-66# 101-66# 101-67 101-67# 101-69 101-69 101-69# 101-69#
101-70 101-70# 101-73 101-73# 101-79 101-79# 101-87 101-87 101-87 101-87 101-87 101-87 101-87 101-87
101-87# 101-88 101-83# 101-91 101-91 101-91# 101-91# 101-92 101-92# 101-114 101-114 101-114# 101-114# 101-114#
101-115 101-115# 101-122 101-122 101-122 101-122 101-122# 101-122# 101-122# 101-122# 101-122# 101-123 101-123# 102-1
102-1# 102-3 102-3 102-3# 102-3# 102-13 102-13 102-13# 102-13# 102-13# 102-14 102-14# 102-15 102-15
102-15# 102-15# 102-15# 102-16 102-16# 102-18 102-18 102-18 102-18 102-18 102-18# 102-18# 102-18# 102-18#
102-24 102-24 102-24 102-24 102-24 102-24 102-24# 102-24# 102-24# 102-24# 102-24# 102-24# 103-10 103-10
103-10# 103-10# 103-10# 103-11 103-11# 106-5 106-5 106-5# 106-5# 106-5# 106-6 106-6# 108-12 108-12
108-12 108-12 108-12 108-12# 108-12# 108-12# 108-12# 108-21 108-21 108-21 108-21 108-21 108-21
108-21 108-21# 108-21# 108-21# 108-21# 108-21# 108-21# 108-21# 108-30 108-30# 108-31 108-31# 108-32 108-32
108-32 108-32 108-32 108-32 108-32# 108-32# 108-32# 108-32# 108-35 108-35# 108-42 108-42 108-42# 108-42#
108-45 108-45 108-45# 108-45# 109-2 109-2 109-2 109-2 109-2# 109-2# 109-2# 109-2# 109-2# 109-3
109-3# 109-6 109-6 109-6 109-6 109-6# 109-6# 109-6# 109-6# 109-6# 109-7 109-7# 109-11 109-11
109-11 109-11 109-11# 109-11# 109-11# 109-11# 109-11# 109-12 109-12# 109-15 109-15# 110-18 110-18# 111-30
111-30# 112-28 112-28# 112-30 112-30 112-30 112-30 112-30 112-30# 112-30# 112-30# 112-30# 112-30#
112-30# 112-34 112-34 112-34# 112-34# 112-37 112-37 112-37 112-37 112-37# 112-37# 112-37# 112-37# 112-37#
112-40 112-40# 112-41 112-41# 112-43 112-43# 112-44 112-44# 112-46 112-46# 112-47 112-47# 112-50 112-50#
112-51 112-51# 112-56 112-56# 112-57 112-57# 112-66 112-66# 113-6 113-6# 114-17 114-17 114-17 114-17
114-17# 114-17# 114-17# 114-17# 114-17# 114-17# 114-27 114-27 114-27 114-27# 114-27# 114-27# 114-27# 114-27#
115-11 115-11# 115-19 115-19 115-19 115-19 115-19# 115-19# 115-19# 115-19# 115-19# 116-16 116-16 116-16
116-16 116-16 116-16 116-16 116-16# 116-16# 116-16# 116-16# 116-16# 116-16# 116-17 116-17 116-17 116-17
116-17 116-17 116-17# 116-17# 116-17# 116-17# 116-17# 116-17# 116-18 116-18 116-18# 116-18# 116-33 116-33#
116-41 116-41 116-41 116-41 116-41# 116-41# 116-41# 116-41# 116-41# 116-44 116-44 116-44# 116-44# 116-55
116-55 116-55# 116-55# 116-61 116-61 116-61 116-61 116-61# 116-61# 116-61# 116-61# 116-61# 116-63 116-63
116-63# 116-63# 116-67 116-67 116-67 116-67 116-67# 116-67# 116-67# 116-67# 116-67# 116-69 116-69 116-69

CROSS REFERENCE TABLE (CREF V01-05)

```
          116-69   16-69   116-69#  116-69#  116-69#  116-69#  116-72   116-72   116-72#  116-72#  116-77   116-77   116-77#  116-77#
          117-24   117-24#  118-14   118-14#  119-12   119-12   119-12   119-12   119-12   119-12#  119-12#  119-12#  119-12#  119-19
          119-19   119-19   119-19   119-19   119-19#  119-19#  119-19#  119-19#  119-24   119-24   119-24   119-24   119-24   119-24#
          119-24#  119-24#  119-24#  119-36   119-36#  119-37   119-37#  119-64   119-64   119-64   119-64   119-64   119-64   119-64
          119-64   119-64#  119-64#  119-64#  119-64#  119-68   119-68   119-68   119-68   119-68   119-68   119-68   119-68   119-68#
          119-68#  119-68#  119-68#  119-76   119-76   119-76   119-76   119-76   119-76#  119-76#  119-76#  119-76#  119-113  119-113#
          119-116  119-116#  119-117  119-117  119-117#  119-117#  120-12   120-12   120-12   120-12   120-12   120-12   120-12   120-12
          120-12#  120-12#  120-12#  120-12#  120-12#  120-12#  120-12#  120-14   120-14   120-14   120-14   120-14   120-14   120-14
          120-14   120-14#  120-14#  120-14#  120-14#  120-23   120-23   120-23   120-23   120-23   120-23   120-23   120-23   120-23#
          120-23#  120-23#  120-23#  120-29   120-29   120-29   120-29   120-29   120-29   120-29#  120-29#  120-29#  120-29#  120-32
          120-32   120-32   120-32   120-32   120-32   120-32#  120-32#  120-32#  120-32#  120-33   120-33   120-33   120-33   120-33
          120-33   120-33#  120-33#  120-33#  120-33#  120-36   120-36   120-36   120-36   120-36   120-36   120-36#  120-36#  120-36#
          120-36#  120-41   120-41   120-41   120-41   120-41   120-41   120-41   120-41   120-41#  120-41#  120-41#  120-41#  120-50
          120-50   120-50   120-50   120-50   120-50   120-50#  120-50#  120-50#  120-50#  120-51   120-51   120-51   120-51   120-51
          120-51   120-51#  120-51#  120-51#  120-51#  120-54   120-54   120-54   120-54   120-54   120-54   120-54#  120-54#  120-54#
          120-54#  120-57   120-57   120-57   120-57   120-57   120-57   120-57#  120-57#  120-57#  120-57#  120-59   120-59   120-59
          120-59   120-59   120-59   120-59#  120-59#  120-59#  120-59#  120-77   120-77   120-77   120-77   120-77   120-77#  120-77#
          120-77#  120-77#  120-78   120-78   120-78   120-78   120-78   120-78#  120-78#  120-78#  120-78#  120-79   120-79   120-79
          120-79   120-79   120-79#  120-79#  120-79#  120-79#  120-80   120-80   120-80   120-80   120-80   120-80#  120-80#  120-80#
          120-80#  120-81   120-81   120-81   120-81   120-81   120-81#  120-81#  120-81#  120-81#  120-82   120-82   120-82   120-82
          120-82   120-82#  120-82#  120-82#  120-82#  120-83   120-83   120-83   120-83   120-83   120-83   120-83   120-83   120-83#
          120-83#  120-83#  120-83#  120-121  120-121  120-121  120-121  120-121  120-121  120-121  120-121  120-121#  120-121#  120-121#
          120-121#  120-127  120-127  120-127  120-127  120-127  120-127  120-127  120-127  120-127#  120-127#  120-127#  120-127#  120-131
          120-131  120-131  120-131  120-131  120-131  120-131  120-131  120-131#  120-131#  120-131#  120-131#  120-146  120-146  120-146
          120-146  120-146  120-146  120-146  120-146  120-146#  120-146#  120-146#  120-146#  120-150  120-150  120-150  120-150  120-150
          120-150  120-150  120-150  120-150#  120-150#  120-150#  120-150#  120-156  120-156  120-156  120-156  120-156  120-156  120-156
          120-156  120-156#  120-156#  120-156#  120-156#  120-160  120-160  120-160  120-160  120-160  120-160  120-160  120-160  120-160#
          120-160#  120-160#  120-160#  120-166  120-166  120-166  120-166  120-166  120-166  120-166#  120-166#  120-166#  120-166#  120-180
          120-180  120-180  120-180  120-180  120-180  120-180  120-180  120-180#  120-180#  120-180#  120-180#  120-184  120-184  120-184
          120-184  120-184  120-184  120-184  120-184  120-184#  120-184#  120-184#  120-184#  120-189  120-189#  121-50  121-50#  122-1
          122-1  122-1  122-1  122-1#  122-2  122-2  122-2  122-2  122-2#  122-3  122-3  122-3  122-3  122-3
          122-3#  122-4  122-4  122-4  122-4  122-4  122-4#  122-5  122-5  122-5  122-5  122-5  122-5#  122-6
          122-6  122-6  122-6#  122-7  122-7#  123-12  123-12#  124-1  124-1  124-1  124-1#  124-2  124-2  124-2#
          124-2#  124-3  124-3  124-3  124-3  124-3  124-3#  124-4  124-4  124-4  124-4  124-4  124-4#  124-5
          124-5  124-5  124-5#  124-6  124-6  124-6  124-6#  124-7  124-7  124-7  124-7#  124-11  124-11#  125-14
          125-14  125-14  125-14#  126-14  126-14  126-14#  126-14#
M$GNLS     41-66   41-66#  108-32  108-32#  119-64  119-64#  119-68  119-68#  120-14  120-14#  120-23  120-23#  120-29  120-29#
          120-32  120-32#  120-33  120-33#  120-36  120-36#  120-41  120-41#  120-50  120-50#  120-51  120-51#  120-54  120-54#
          120-57  120-57#  120-59  120-59#  120-83  120-83#  120-121  120-121#  120-127  120-127#  120-131  120-131#  120-146  120-146#
          120-150  120-150#  120-156  120-156#  120-160  120-160#  120-166  120-166#  120-180  120-180#  120-184  120-184#
M$GNSU    112-28   112-28#  112-41  112-41#  112-44  112-44#  112-47  112-47#  112-51  112-51#  112-57  112-57#
M$GNTA     7-27    7-27#    8-27    8-27#   27-17   27-17#   27-22   27-22#   27-28   27-28#   27-32   27-32#   27-37   27-37#
          27-49   27-49#   27-54   27-54#   27-59   27-59#   27-63   27-63#   27-67   27-67#   27-71   27-71#   27-75   27-75#
          27-79   27-79#   27-83   27-83#   67-48   67-48#   81-14   81-14#   82-22   82-22#   95-9    95-9#    96-3    96-3#
          99-156  99-156#  109-15  109-15#  110-18  110-18#  111-30  111-30#  112-40  112-40#  112-43  112-43#  112-46  112-46#
          112-50  112-50#  112-56  112-56#  112-66  112-66#  113-6   113-6#   117-24  117-24#  118-14  118-14#  120-189  120-189#
          122-7   122-7#   124-11  124-11#  126-14  126-14#  126-21  126-21#
M$GNTE    112-18  112-18#  117-3   117-3#   118-3   118-3#   119-3   119-3#
M$HAPT     5-77    5-77#
M$HNAP     5-77    5-77#
M$INCR     5-43    5-43#    7-10    7-10    7-10#    7-10#    8-10    8-10    8-10#    8-10#    9-48    9-48#   27-14   27-14
          27-14#  27-14#  27-15#  27-16#  27-17#  27-19   27-19   27-19#  27-19#  27-20#  27-21#  27-22#  27-24   27-24
          27-24#  27-24#  27-25#  27-26#  27-27#  27-28#  27-30   27-30   27-30#  27-30#  27-31#  27-32#  27-34   27-34
          27-34#  27-34#  27-35#  27-36#  27-37#  27-39   27-39   27-39#  27-39#  27-40#  27-41#  27-42#  27-45#  27-49#
          27-51   27-51   27-51#  27-51#  27-52#  27-53#  27-54#  27-56   27-56   27-56#  27-56#  27-57#  27-58#  27-59#
          27-61   27-61   27-61#  27-61#  27-62#  27-63#  27-65   27-65   27-65#  27-65#  27-66#  27-67#  27-69   27-69
          27-69#  27-69#  27-70#  27-71#  27-73   27-73   27-73#  27-73#  27-74#  27-75#  27-77   27-77   27-77#  27-77#
```

CROSS REFERENCE TABLE (CREF V01-05)

	27-78#	27-79#	27-81	27-81	27-81#	27-81#	27-82#	27-83#	28-7#	28-8#	32-20#	32-21#	34-22#	34-40#
	34-43#	34-54#	35-11#	35-18#	35-25#	36-6#	41-32#	41-34#	41-42#	41-43#	41-44#	41-45#	41-53#	41-59#
	41-62#	41-66	41-66#	41-66#	41-113#	49-28#	49-33#	49-34#	51-27#	51-60#	51-63#	52-41#	52-44#	53-21#
	55-18#	56-6#	56-14#	56-43#	56-56#	56-64#	58-13#	58-20#	59-24#	61-59#	61-69#	64-39#	64-43#	65-4#
	65-13#	65-30#	67-27#	67-31	67-?1	67-31#	67-31#	67-34#	67-37#	67-48#	68-32#	71-3#	75-28#	75-36#
	75-44#	77-17#	77-36#	81-10	81-10	81-10#	81-10#	82-18	82-18	82-18#	82-18#	85-18#	85-37#	86-13#
	86-24#	86-27#	86-30#	87-44#	88-27#	88-36#	89-5#	92-12#	93-21#	93-24#	93-26#	93-31#	93-41#	93-76#
	93-83#	93-89#	93-99#	93-102#	94-1#	94-2#	94-3#	94-4#	95-5	95-5	95-5#	95-5#	96-1	96-1
	96-1#	96-1#	97-10#	97-11#	98-23#	98-24#	99-38	99-38#	99-45	99-45	99-45#	99-45#	99-59#	99-70#
	99-71#	99-74#	99-75#	99-88#	99-107#	99-108#	99-156#	100-8	100-8	100-8#	100-8#	101-8	101-8	101-8#
	101-8#	101-63#	101-66#	101-69#	101-73#	101-79#	101-88#	101-91#	101-114#	101-122#	101-123#	102-1#	102-3#	102-13#
	102-15#	102-18#	102-24#	103-10#	106-5#	108-12#	108-21#	108-30#	108-32	108-32#	108-32#	108-35#	108-42#	108-45#
	109-2#	109-3#	109-6#	109-7#	109-11#	109-12#	109-15#	110-10	110-10	110-10#	110-10#	110-18#	111-8	111-8
	111-8#	111-8#	111-30#	112-16	112-16#	112-18	112-18	112-18	112-18#	112-18#	112-18#	112-28	112-28	112-28
	112-28#	112-28#	112-28#	112-30#	112-34#	112-37#	112-40#	112-41	112-41	112-41	112-41#	112-41#	112-41#	112-43#
	112-44	112-44	112-44	112-44#	112-44#	112-44#	112-46#	112-47	112-47	112-47	112-47#	112-47#	112-47#	112-50#
	112-51	112-51	112-51	112-51#	112-51#	112-51#	112-56#	112-57	112-57	112-57	112-57#	112-57#	112-57#	112-66#
	113-6#	114-17#	114-27#	115-11#	115-19#	116-16#	116-17#	116-18#	116-33#	116-41#	116-44#	116-55#	116-61#	116-63#
	116-67#	116-69#	116-72#	116-77#	117-3	117-3	117-3	117-3#	117-3#	117-3#	117-24#	118-3	118-3	118-3
	118-3#	118-3#	118-3#	118-14#	119-3	119-3	119-3	119-3#	119-3#	119-3#	119-12#	119-19#	119-24#	119-36#
	119-64	119-64#	119-64#	119-68	119-68#	119-68#	119-76#	119-113#	119-116#	119-117#	120-12#	120-14	120-14#	120-14#
	120-23.	120-23#	120-23#	120-29	120-29#	120-29#	120-32	120-32#	120-32#	120-33	120-33#	120-33#	120-36	120-36#
	120-36#	120-41	120-41#	120-41#	120-50	120-50#	120-50#	120-51	120-51#	120-51#	120-54	120-54#	120-54#	120-57
	120-57#	120-57#	120-59	120-59#	120-59#	120-77#	120-78#	120-79#	120-80#	120-81#	120-82#	120-83	120-83#	120-83#
	120-121	120-121#	120-121#	120-127	120-127#	120-127#	120-131	120-131#	120-131#	120-146	120-146#	120-146#	120-150	120-150#
	120-150#	120-156	120-156#	120-156#	120-160	120-160#	120-160#	120-166	120-166#	120-166#	120-180	120-180#	120-180#	120-184
	120-184#	120-184#	120-189#	121-39	121-39#	121-50	121-50	121-50#	121-50#	123-12	123-12	123-12#	123-12#	126-13
	126-13#	126-14	126-14	126-14	126-14#									
M$LDRO	77-36	77-36#	86-27	86-27#	93-21	93-21#	101-63	101-63#	101-66	101-66#	101-69	101-69#	101-91	101-91#
	101-114	101-114#	102-13	102-13#	102-15	102-15#	103-10	103-10#	106-5	106-5#	108-42	108-42#	112-34	112-34#
	116-18	116-18#	116-44	116-44#	116-55	116-55#	116-63	116-63#	116-72	116-72#	116-77	116-77#		
M$MCHI	5-16	5-16#												
M$MCLO	5-16	5-16#												
M$POP	7-27	7-27#	8-27	8-27#	8-29	8-29#	27-17	27-17#	27-22	27-22#	27-28	27-28#	27-32	27-32#
	27-37	27-37#	27-43	27-49#	27-54	27-54#	27-59	27-59#	27-63	27-63#	27-67	27-67#	27-71	27-71#
	27-75	27-75#	27-79	27-79#	27-83	27-83#	67-48	67-48#	81-14	81-14#	82-22	82-22#	95-9	95-9#
	96-3	96-3#	98-28	98-28#	99-156	99-156#	100-14	100-14#	109-15	109-15#	110-18	110-18#	111-30	111-30#
	111-31	111-31#	112-40	112-40#	112-43	112-43#	112-46	112-46#	112-50	112-50#	112-56	112-56#	112-66	112-66#
	113-6	113-6#	117-24	117-24#	118-14	118-14#	120-189	120-189#	120-190	120-190#	122-7	122-7#	124-11	124-11#
	125-15	125-15#												
M$PRIN	27-15	27-15#	27-16	27-16#	27-20	27-20#	27-21	27-21#	27-25	27-25#	27-26	27-26#	27-27	27-27#
	27-31	27-31#	27-35	27-35#	27-36	27-36#	27-40	27-40#	27-41	27-41#	27-42	27-42#	27-45	27-45#
	27-52	27-52#	27-53	27-53#	27-57	27-57#	27-58	27-58#	27-62	27-62#	27-66	27-66#	27-70	27-70#
	27-74	27-74#	27-78	27-78#	27-82	27-82#	41-34	41-34#	41-42	41-42#	41-43	41-43#	41-44	41-44#
	41-45	41-45#	41-53	41-53#	41-59	41-59#	41-62	41-62#	41-113	41-113#	49-33	49-33#	49-34	49-34#
	51-60	51-60#	51-63	51-63#	52-44	52-44#	53-21	53-21#	55-18	55-18#	56-6	56-6#	56-14	56-14#
	56-43	56-43#	56-56	56-56#	56-64	56-64#	58-13	58-13#	58-20	58-20#	61-59	61-59#	61-69	61-69#
	64-39	64-39#	64-43	64-43#	65-4	65-4#	65-13	65-13#	65-30	65-30#	67-34	67-34#	67-37	67-37#
	98-23	98-23#	98-24	98-24#	99-59	99-59#	99-70	99-70#	99-71	99-71#	99-74	99-74#	99-75	99-75#
	99-88	99-88#	99-107	99-107#	99-108	99-108#	102-18	102-18#	108-12	108-12#	108-21	108-21#	116-16	116-16#
	116-69	116-69#	119-12	119-12#	119-19	119-19#	119-24	119-24#	119-76	119-76#	120-12	120-12#	120-77	120-77#
	120-78	120-78#	120-79	120-79#	120-80	120-80#	120-81	120-81#	120-82	120-82#				
M$PUSH	5-43	5-43#	7-10	7-10#	8-10	8-10#	9-48	9-48#	27-14	27-14#	27-19	27-19#	27-24	27-24#
	27-30	27-30#	27-34	27-34#	27-39	27-39#	27-51	27-51#	27-56	27-56#	27-61	27-61#	27-65	27-65#
	27-69	27-69#	27-73	27-73#	27-77	27-77#	27-81	27-81#	67-31	67-31#	81-10	81-10#	82-18	82-18#
	95-5	95-5#	96-1	96-1#	99-38	99-38#	99-45	99-45#	100-8	100-8#	101-8	101-8#	110-10	110-10#
	111-8	111-8#	112-16	112-16#	112-18	112-18#	112-28	112-28#	112-41	112-41#	112-44	112-44#	112-47	112-47#

	112-51	112-51#	112-57	112-57#	117-3	117-3#	118-3	118-3#	119-3	119-3#	121-39	121-39#	121-50	121-50#
	123-12	123-12#												
M$PUT	27-15	27-15	27-15#	27-16	27-16	27-16	27-16#	27-20	27-20	27-20#	27-21	27-21	27-21	27-21#
	27-25	27-25	27-25#	27-26	27-26	27-26	27-26#	27-27	27-27	27-27	27-27#	27-31	27-31	27-31
	27-31#	27-35	27-35	27-35#	27-36	27-36	27-36	27-36#	27-40	27-40	27-40#	27-41	27-41	27-41#
	27-42	27-42	27-42#	27-45	27-45	27-45	27-45	27-45#	27-52	27-52	27-52	27-52#	27-53	27-53
	27-53#	27-57	27-57	27-57	27-57#	27-58	27-58	27-58#	27-62	27-62	27-62	27-62	27-62#	27-66
	27-66	27-66	27-66	27-66#	27-70	27-70	27-70	27-70#	27-74	27-74	27-74	27-74#	27-78	27-78
	27-78	27-78#	27-82	27-82	27-82	27-82	27-82#	41-34	41-34	41-34#	41-42	41-42	41-42	41-42
	41-42	41-42#	41-43	41-43	41-43#	41-44	41-44	41-44#	41-45	41-45	41-45#	41-53	41-53	41-53
	41-53#	41-59	41-59	41-59	41-59#	41-62	41-62	41-62#	41-113	41-113	41-113#	49-33	49-33	49-33
	49-33	49-33	49-33#	49-34	49-34	49-34#	51-60	51-60	51-60#	51-63	51-63	51-63#	52-44	52-44
	52-44	52-44#	53-21	53-21	53-21	53-21	53-21	53-21#	55-18	55-18	55-18#	56-6	56-6	56-6
	56-6#	56-14	56-14	56-14	56-14#	56-43	56-43	56-43#	56-56	56-56	56-56#	56-64	56-64	56-64
	56-64#	58-13	58-13	58-13#	58-20	58-20	58-20#	61-59	61-59	61-59	61-59#	61-69	61-69	61-69
	61-69#	64-39	64-39	64-39#	64-43	64-43	64-43#	65-4	65-4	65-4	65-4#	65-13	65-13	65-13
	65-13#	65-30	65-30	65-30	65-30#	67-34	67-34	67-34	67-34	67-34	67-34#	67-37	67-37	67-37
	67-37#	68-32	68-32	68-32	68-32	68-32#	77-17	77-17	77-17	77-17	77-17#	86-24	86-24	86-24
	86-24	86-24#	98-23	98-23	98-23	98-23	98-23	98-23#	98-24	98-24	98-24#	99-59	99-59	99-59
	99-59#	99-70	99-71	99-70	99-70	99-70	99-70#	99-71	99-71	99-71#	99-74	99-74	99-74#	99-75
	99-75	99-75#	99-86	99-88	99-88	99-88	99-88#	99-107	99-107	99-107	99-107	99-107	99-107	99-107#
	99-108	99-108	99-108	99-108	99-108	99-108#	102-18	102-18	102-18#	102-24	102-24	102-24	102-24	102-24#
	108-12	108-12	108-12#	108-21	108-21	108-21	108-21	108-21	108-21#	112-30	112-30	112-30	112-30	112-30#
	116-16	116-16	116-16	116-16	116-16#	116-17	116-17	116-17	116-17	116-17#	116-69	116-69	116-69#	119-12
	119-12	119-12#	119-19	119-19	119-19#	119-24	119-24	119-24#	119-76	119-76	119-76#	120-12	120-12	120-12
	120-12	120-12	120-12#	120-77	120-77	120-77#	120-78	120-78	120-78#	120-79	120-79	120-79#	120-80	120-80
	120-80#	120-81	120-81	120-81#	120-82	120-82	120-82#							
M$PUT1	27-15	27-15	27-15#	27-15#	27-16	27-16	27-16	27-16#	27-16#	27-16#	27-20	27-20	27-20#	27-20#
	27-21	27-21	27-21	27-21#	27-21#	27-21#	27-25	27-25	27-25#	27-25#	27-26	27-26	27-26	27-26#
	27-26#	27-26#	27-27	27-27	27-27	27-27#	27-27#	27-27#	27-31	27-31	27-31	27-31#	27-31#	27-31#
	27-35	27-35	27-35#	27-35#	27-36	27-36	27-36	27-36#	27-36#	27-36#	27-40	27-40	27-40#	27-40#
	27-41	27-41	27-41#	27-41#	27-42	27-42	27-42#	27-42#	27-45	27-45	27-45	27-45	27-45#	27-45#
	27-45#	27-45#	27-52	27-52	27-52	27-52#	27-52#	27-52#	27-53	27-53	27-53#	27-53#	27-57	27-57
	27-57	27-57#	27-57#	27-57#	27-58	27-58	27-58#	27-58#	27-62	27-62	27-62	27-62	27-62#	27-62#
	27-62#	27-62#	27-66	27-66	27-66	27-66	27-66#	27-66#	27-66#	27-66#	27-70	27-70	27-70	27-70#
	27-70#	27-70#	27-74	27-74	27-74	27-74#	27-74#	27-74#	27-78	27-78	27-78	27-78#	27-78#	27-78#
	27-82	27-82	27-82	27-82	27-82#	27-82#	27-82#	27-82#	41-34	41-34	41-34#	41-34#	41-42	41-42
	41-42	41-42	41-42	41-42#	41-42#	41-42#	41-42#	41-42#	41-43	41-43	41-43#	41-43#	41-44	41-44
	41-44#	41-44#	41-45	41-45	41-45#	41-45#	41-53	41-53	41-53	41-53#	41-53#	41-53#	41-59	41-59
	41-59	41-59#	41-59#	41-59#	41-62	41-62	41-62#	41-62#	41-113	41-113	41-113#	41-113#	49-33	49-33
	49-33	49-33	49-33	49-33#	49-33#	49-33#	49-33#	49-33#	49-34	49-34	49-34#	49-34#	51-60	51-60
	51-60#	51-60#	51-63	51-63	51-63#	51-63#	52-44	52-44	52-44	52-44#	52-44#	52-44#	53-21	53-21
	53-21	53-21	53-21	53-21#	53-21#	53-21#	53-21#	53-21#	55-18	55-18	55-18#	55-18#	56-6	56-6
	56-6	56-6#	56-6#	56-6#	56-14	56-14	56-14	56-14#	56-14#	56-14#	56-43	56-43	56-43#	56-43#
	56-56	56-56	56-56#	56-56#	56-64	56-64	56-64	56-64#	56-64#	56-64#	58-13	58-13	58-13#	58-13#
	58-20	58-20	58-20#	58-20#	61-59	61-59	61-59	61-59#	61-59#	61-59#	61-69	61-69	61-69	61-69#
	61-69#	61-69#	64-39	64-39	64-39#	64-39#	64-43	64-43	64-43#	64-43#	65-4	65-4	65-4	65-4#
	65-4#	65-4#	65-13	65-13	65-13	65-13#	65-13#	65-13#	65-30	65-30	65-30	65-30#	65-30#	65-30#
	67-34	67-34	67-34	67-34	67-34	67-34#	67-34#	67-34#	67-34#	67-34#	67-37	67-37	67-37	67-37#
	67-37#	67-37#	68-32	68-32	68-32	68-32	68-32#	68-32#	68-32#	68-32#	77-17	77-17	77-17	77-17
	77-17#	77-17#	77-17#	77-17#	86-24	86-24	86-24	86-24	86-24#	86-24#	86-24#	86-24#	98-23	98-23
	98-23	98-23	98-23	98-23#	98-23#	98-23#	98-23#	98-23#	98-24	98-24	98-24#	98-24#	99-59	99-59
	99-59	99-59#	99-59#	99-59#	99-70	99-70	99-70	99-70	99-70	99-70#	99-70#	99-70#	99-70#	99-70#
	99-71	99-71	99-71#	99-71#	99-74	99-74	99-74#	99-74#	99-75	99-75	99-75#	99-75#	99-88	99-88
	99-88	99-88	99-88#	99-88#	99-88#	99-88#	99-107	99-107	99-107	99-107	99-107	99-107	99-107#	99-107#
	99-107#	99-107#	99-107#	99-107#	99-108	99-108	99-108	99-108	99-108	99-108#	99-108#	99-108#	99-108#	99-108#
	102-18	102-18	102-18#	102-18#	102-24	102-24	102-24	102-24	102-24#	102-24#	102-24#	102-24#	108-12	108-12

CROSS REFERENCE TABLE (CREF V01-05)

	108-12#	108-12#	108-21	108-21	108-21	108-21	108-21	108-21#	108-21#	108-21#	108-21#	108-21#	112-30	112-30
	112-30	112-30	112-30#	112-30#	112-30#	112-30#	116-16	116-16	116-16	116-16	116-16#	116-16#	116-16#	116-16#
	116-17	116-17	116-17	116-17	116-17#	116-17#	116-17#	116-17#	116-69	116-69	116-69#	116-69#	119-12	119-12
	119-12#	119-12#	119-19	119-19	119-19#	119-19#	119-24	119-24	119-24#	119-24#	119-76	119-76	119-76#	119-76#
	120-12	120-12	120-12	120-12	120-12	120-12#	120-12#	120-12#	120-12#	120-12#	120-77	120-77	120-77#	120-77#
	120-78	120-78	120-78#	120-78#	120-79	120-79	120-79#	120-79#	120-80	120-80	120-80#	120-80#	120-81	120-81
	120-81#	120-81#	120-82	120-82	120-82#	120-82#								
M$RADI	41-66	41-66#	108-32	108-32#	119-64	119-64#	119-68	119-68#	120-14	120-14#	120-23	120-23#	120-29	120-29#
	120-32	120-32#	120-33	120-33#	120-36	120-36#	120-41	120-41#	120-50	120-50#	120-51	120-51#	120-54	120-54#
	120-57	120-57#	120-59	120-59#	120-83	120-83#	120-121	120-121#	120-127	120-127#	120-131	120-131#	120-146	120-146#
	120-150	120-150#	120-156	120-156#	120-160	120-160#	120-166	120-166#	120-180	120-180#	120-184	120-184#	122-1	122-1#
	122-2	122-2#	122-3	122-3#	122-4	122-4#	122-5	122-5#	122-6	122-6#	124-1	124-1#	124-3	124-3#
	124-4	124-4#	124-5	124-5#	124-6	124-6#	124-7	124-7#						
M$RBRO	93-26	93-26#	93-31	93-31#	93-41	93-41#	93-76	93-76#	93-83	93-83#	93-89	93-89#	93-99	93-99#
	93-102	93-102#												
M$RNRO	49-28	49-28#	51-27	51-27#	52-41	52-41#	101-114	101-114#	102-3	102-3#	102-13	102-13#	102-15	102-15#
	103-10	103-10#	106-5	106-5#										
M$SETS	5-43	5-43#	7-10	7-10#	8-10	8-10#	9-48	9-48#	27-14	27-14#	27-19	27-19#	27-24	27-24#
	27-30	27-30#	27-34	27-34#	27-39	27-39#	27-51	27-51#	27-56	27-56#	27-61	27-61#	27-65	27-65#
	27-69	27-69#	27-73	27-73#	27-77	27-77#	27-81	27-81#	67-31	67-31#	81-10	81-10#	82-18	82-18#
	95-5	95-5#	96-1	96-1#	99-38	99-38#	99-45	99-45#	100-8	100-8#	101-8	101-8#	110-10	110-10#
	111-8	111-8#	112-16	112-16#	112-18	112-18#	112-28	112-28#	112-41	112-41#	112-44	112-44#	112-47	112-47#
	112-51	112-51#	112-57	112-57#	117-3	117-3#	118-3	118-3#	119-3	119-3#	121-39	121-39#	121-50	121-50#
	123-12	123-12#												
M$SVC	27-15	27-15#	27-16	27-16#	27-17	27-17#	27-20	27-20#	27-21	27-21#	27-22	27-22#	27-25	27-25#
	27-26	27-26#	27-27	27-27#	27-28	27-28#	27-31	27-31#	27-32	27-32#	27-35	27-35#	27-36	27-36#
	27-37	27-37#	27-40	27-40#	27-41	27-41#	27-42	27-42#	27-45	27-45#	27-49	27-49#	27-52	27-52#
	27-53	27-53#	27-54	27-54#	27-57	27-57#	27-58	27-58#	27-59	27-59#	27-62	27-62#	27-63	27-63#
	27-66	27-66#	27-67	27-67#	27-70	27-70#	27-71	27-71#	27-74	27-74#	27-75	27-75#	27-78	27-78#
	27-79	27-79#	27-82	27-82#	27-83	27-83#	28-7	28-8	28-8#	32-20	32-21	32-21#	34-22	34-40
	34-43	34-43#	34-54	34-54#	35-11	35-18	35-25	36-6	41-32	41-32#	41-34	41-34#	41-42	41-42#
	41-43	41-43#	41-44	41-44#	41-45	41-45#	41-53	41-53#	41-59	41-59#	41-62	41-62#	41-66	41-66#
	41-113	41-113#	49-28	49-28#	49-33	49-33#	49-34	49-34#	51-27	51-27#	51-60	51-60#	51-63	51-63#
	52-41	52-41#	52-44	52-44#	53-21	53-21#	55-18	55-18#	56-6	56-6#	56-14	56-14#	56-43	56-43#
	56-56	56-56#	56-6.	56-64#	58-13	58-13#	58-20	58-20#	59-24	61-59	61-59#	61-69	61-69#	64-39
	64-39#	64-43	64-43#	65-4	65-4#	65-13	65-13#	65-30	65-30#	67-27	67-27#	67-34	67-34#	67-37
	67-37#	67-48	67-48#	68-32	68-32#	71-3	75-28	75-28#	75-36	75-44	77-17	77-17#	77-36	77-36#
	85-18	85-37	86-13	86-13#	86-24	86-24#	86-27	86-27#	86-30	87-44	88-27	88-27#	88-36	89-5
	92-12	92-12#	93-21	93-21#	93-24	93-24#	93-26	93-26#	93-31	93-31#	93-41	93-41#	93-76	93-76#
	93-83	93-83#	93-89	93-89#	93-99	93-99#	93-102	93-102#	94-1	94-2	94-2#	94-3	94-4	94-4#
	97-10	97-10#	97-11	97-11#	98-23	98-23#	98-24	98-24#	99-59	99-59#	99-70	99-70#	99-71	99-71#
	99-74	99-74#	99-75	99-75#	99-88	99-88#	99-107	99-107#	99-108	99-108#	99-118#	99-156	99-156#	101-63
	101-63#	101-66	101-66#	101-69	101-69#	101-73	101-73#	101-79	101-79#	101-88	101-88#	101-91	101-91#	101-114
	101-114#	101-122	101-123	101-123#	102-1	102-1#	102-3	102-3#	102-13	102-13#	102-15	102-15#	102-18	102-18#
	102-24	102-24#	103-10	103-10#	106-5	106-5#	108-12	108-12#	108-21	108-21#	108-30	108-30#	108-32	108-32#
	108-35	108-35#	108-42	108-42#	108-45	108-45#	109-2	109-3	109-3#	109-6	109-7	109-7#	109-11	109-12
	109-12#	109-15	109-15#	110-18	110-18#	111-30	111-30#	112-28	112-28#	112-30	112-30#	112-34	112-34#	112-37
	112-40	112-40#	112-41	112-41#	112-43	112-43#	112-44	112-44#	112-46	112-46#	112-47	112-47#	112-50	112-50#
	112-51	112-51#	112-56	112-56#	112-57	112-57#	112-66	112-66#	113-6	113-6#	114-17	114-27	115-11	115-11#
	115-19	116-16	116-16#	116-17	116-17#	116-18	116-18#	116-33	116-33#	116-41	116-44	116-44#	116-55	116-55#
	116-61	116-63	116-63#	116-67	116-69	116-69#	116-72	116-72#	116-77	116-77#	117-24	117-24#	118-14	118-14#
	119-12	119-12#	119-19	119-19#	119-24	119-24#	119-36	119-36#	119-64	119-64#	119-68	119-68#	119-76	119-76#
	119-113	119-113#	119-116	119-116#	119-117	119-117#	120-12	120-12#	120-14	120-14#	120-23	120-23#	120-29	120-29#
	120-32	120-32#	120-33	120-33#	120-36	120-36#	120-41	120-41#	120-50	120-50#	120-51	120-51#	120-54	120-54#
	120-57	120-57#	120-59	120-59#	120-77	120-77#	120-78	120-78#	120-79	120-79#	120-80	120-80#	120-81	120-81#
	120-82	120-82#	120-83	120-83#	120-121	120-121#	120-127	120-127#	120-131	120-131#	120-146	120-146#	120-150	120-150#
	120-156	120-156#	120-160	120-160#	120-166	120-166#	120-180	120-180#	120-184	120-184#	120-189	120-189#		

CROSS REFERENCE TABLE (CREF V01-05)

M$TLAB	27-15#	27-16#	27-17#	27-20#	27-21#	27-22#	27-25#	27-26#	27-27#	27-28#	27-31#	27-32#	27-35#	27-36#
	27-37#	27-40#	27-41#	27-42#	27-45#	27-49#	27-52#	27-53#	27-54#	27-57#	27-58#	27-59#	27-62#	27-63#
	27-66#	27-67#	27-70#	27-71#	27-74#	27-75#	27-78#	27-79#	27-82#	27-83#	28-7#	28-8#	32-20#	32-21#
	34-22#	34-40#	34-43#	34-54#	35-11#	35-18#	35-25#	36-6#	41-32#	41-34#	41-42#	41-43#	41-44#	41-45#
	41-53#	41-59#	41-62#	41-66#	41-113#	49-28#	49-33#	49-34#	51-27#	51-60#	51-63#	52-41#	52-44#	53-21#
	55-18#	56-6#	56-14#	56-43#	56-56#	56-64#	58-13#	58-20#	59-24#	61-59#	61-69#	64-39#	64-43#	65-4#
	65-13#	65-30#	67-27#	67-34#	67-37#	67-48#	68-32#	71-3#	75-28#	75-36#	75-44#	77-17#	77-36#	85-18#
	85-37#	86-13#	86-24#	86-27#	86-30#	87-44#	88-27#	88-36#	89-5#	92-12#	93-21#	93-24#	93-26#	93-31#
	93-41#	93-76#	93-83#	93-89#	93-99#	93-102#	94-1#	94-2#	94-3#	94-4#	97-10#	97-11#	98-23#	98-24#
	99-59#	99-70#	99-71#	99-74#	99-75#	99-88#	99-107#	99-108#	99-156#	101-63#	101-66#	101-69#	101-73#	101-79#
	101-88#	101-91#	101-114#	101-122#	101-123#	102-1#	102-3#	102-13#	102-15#	102-18#	102-24#	103-10#	106-5#	108-12#
	108-21#	108-30#	108-32#	108-35#	108-42#	108-45#	109-2#	109-3#	109-6#	109-7#	109-11#	109-12#	109-15#	110-18#
	111-30#	112-28#	112-30#	112-34#	112-37#	112-40#	112-41#	112-43#	112-44#	112-46#	112-47#	112-50#	112-51#	112-56#
	112-57#	112-66#	113-6#	114-17#	114-27#	115-11#	115-19#	116-16#	116-17#	116-18#	116-33#	116-41#	116-44#	116-55#
	116-61#	116-63#	116-67#	116-69#	116-72#	116-77#	117-24#	118-14#	119-12#	119-19#	119-24#	119-36#	119-64#	119-68#
	119-76#	119-113#	119-116#	119-117#	120-12#	120-14#	120-23#	120-29#	120-32#	120-33#	120-36#	120-41#	120-50#	120-51#
	120-54#	120-57#	120-59#	120-77#	120-78#	120-79#	120-80#	120-81#	120-82#	120-83#	120-121#	120-127#	120-131#	120-146#
	120-150#	120-156#	120-160#	120-166#	120-180#	120-184#	120-189#							
M$TSTL	27-15	27-15#	27-16	27-16#	27-17	27-17#	27-20	27-20#	27-21	27-21#	27-22	27-22#	27-25	27-25#
	27-26	27-26#	27-27	27-27#	27-28	27-28#	27-31	27-31#	27-32	27-32#	27-35	27-35#	27-36	27-36#
	27-37	27-37#	27-40	27-40#	27-41	27-41#	27-42	27-42#	27-45	27-45#	27-49	27-49#	27-52	27-52#
	27-53	27-53#	27-54	27-54#	27-57	27-57#	27-58	27-58#	27-59	27-59#	27-62	27-62#	27-63	27-63#
	27-66	27-66#	27-67	27-67#	27-70	27-70#	27-71	27-71#	27-74	27-74#	27-75	27-75#	27-78	27-78#
	27-79	27-79#	27-82	27-82#	27-83	27-83#	28-7	28-7#	28-7#	28-8	28-8#	32-20	32-20#	32-20#
	32-21	32-21#	34-22	34-22#	34-22#	34-40	34-40#	34-40#	34-43	34-43#	34-54	34-54#	35-11	35-11#
	35-11#	35-18	35-18#	35-18#	35-25	35-25#	35-25#	36-6	36-6#	36-6#	41-32	41-32#	41-34	41-34#
	41-42	41-42#	41-43	41-43#	41-44	41-44#	41-45	41-45#	41-53	41-53#	41-59	41-59#	41-62	41-62#
	41-66	41-66#	41-113	41-113#	49-28	49-28#	49-33	49-33#	49-34	49-34#	51-27	51-27#	51-60	51-60#
	51-63	51-63#	52-41	52-41#	52-44	52-44#	53-21	53-21#	55-18	55-18#	56-6	56-6#	56-14	56-14#
	56-43	56-43#	56-56	56-56#	56-64	56-64#	58-13	58-13#	58-20	58-20#	59-24	59-24#	59-24#	61-59
	61-59#	61-69	61-69#	64-39	64-39#	64-43	64-43#	65-4	65-4#	65-13	65-13#	65-30	65-30#	67-27
	67-27#	67-34	67-34#	67-37	67-37#	67-48	67-48#	68-32	68-32#	71-3	71-3#	71-3#	75-28	75-28#
	75-36	75-36#	75-36#	75-44	75-44#	75-44#	77-17	77-17#	77-36	77-36#	85-18	85-18#	85-18#	85-37
	85-37#	85-37#	86-13	86-13#	86-24	86-24#	86-27	86-27#	86-30	86-30#	86-30#	87-44	87-44#	87-44#
	88-27	88-27#	88-3?	88-36#	88-36#	89-5	89-5#	89-5#	92-12	92-12#	93-21	93-21#	93-24	93-24#
	93-26	93-26#	93-31	93-31#	93-41	93-41#	93-76	93-76#	93-83	93-83#	93-89	93-89#	93-99	93-99#
	93-102	93-102#	94-1	94-1#	94-1#	94-2	94-2#	94-3	94-3#	94-3#	94-4	94-4#	97-10	97-10#
	97-11	97-11#	98-23	98-23#	98-24	98-24#	99-59	99-59#	99-70	99-70#	99-71	99-71#	99-74	99-74#
	99-75	99-75#	99-88	99-88#	99-107	99-107#	99-108	99-108#	99-156	99-156#	101-63	101-63#	101-66	101-66#
	101-69	101-69#	101-73	101-73#	101-79	101-79#	101-88	101-88#	101-91	101-91#	101-114	101-114#	101-122	101-122#
	101-122#	101-123	101-123#	102-1	102-1#	102-3	102-3#	102-13	102-13#	102-15	102-15#	102-18	102-18#	102-24
	102-24#	103-10	103-10#	106-5	106-5#	108-12	108-12#	108-21	108-21#	108-30	108-30#	108-32	108-32#	108-35
	108-35#	108-42	108-42#	108-45	108-45#	109-2	109-2#	109-2#	109-3	109-3#	109-6	109-6#	109-6#	109-7
	109-7#	109-11	109-11#	109-11#	109-12	109-12#	109-15	109-15#	110-18	110-18#	111-30	111-30#	112-28	112-28#
	112-30	112-30#	112-34	112-34#	112-37	112-37#	112-37#	112-40	112-40#	112-41	112-41#	112-43	112-43#	112-44
	112-44#	112-46	112-46#	112-47	112-47#	112-50	112-50#	112-51	112-51#	112-56	112-56#	112-57	112-57#	112-66
	112-66#	113-6	113-6#	114-17	114-17#	114-17#	114-27	114-27#	114-27#	115-11	115-11#	115-19	115-19#	115-19#
	116-16	116-16#	116-17	116-17#	116-18	116-18#	116-33	116-33#	116-41	116-41#	116-41#	116-44	116-44#	116-55
	116-55#	116-61	116-61#	116-61#	116-63	116-63#	116-67	116-67#	116-67#	116-69	116-69#	116-72	116-72#	116-77
	116-77#	117-24	117-24#	118-14	118-14#	119-12	119-12#	119-19	119-19#	119-24	119-24#	119-36	119-36#	119-64
	119-64#	119-68	119-68#	119-76	119-76#	119-113	119-113#	119-116	119-116#	119-117	119-117#	120-12	120-12#	120-14
	120-14#	120-23	120-23#	120-29	120-29#	120-32	120-32#	120-33	120-33#	120-36	120-36#	120-41	120-41#	120-50
	120-50#	120-51	120-51#	120-54	120-54#	120-57	120-57#	120-59	120-59#	120-77	120-77#	120-78	120-78#	120-79
	120-79#	120-80	120-80#	120-81	120-81#	120-82	120-82#	120-83	120-83#	120-121	120-121#	120-127	120-127#	120-131
	120-131#	120-146	120-146#	120-150	120-150#	120-156	120-156#	120-160	120-160#	120-166	120-166#	120-180	120-180#	120-184
	120-184#	120-189	120-189#											
M$WORD	5-77	5-77#	6-8	6-8	6-8	6-8	6-8	6-8#	28-7	28-7	28-7	28-7#	32-20	32-20

CROSS REFERENCE TABLE (CREF V01-05)