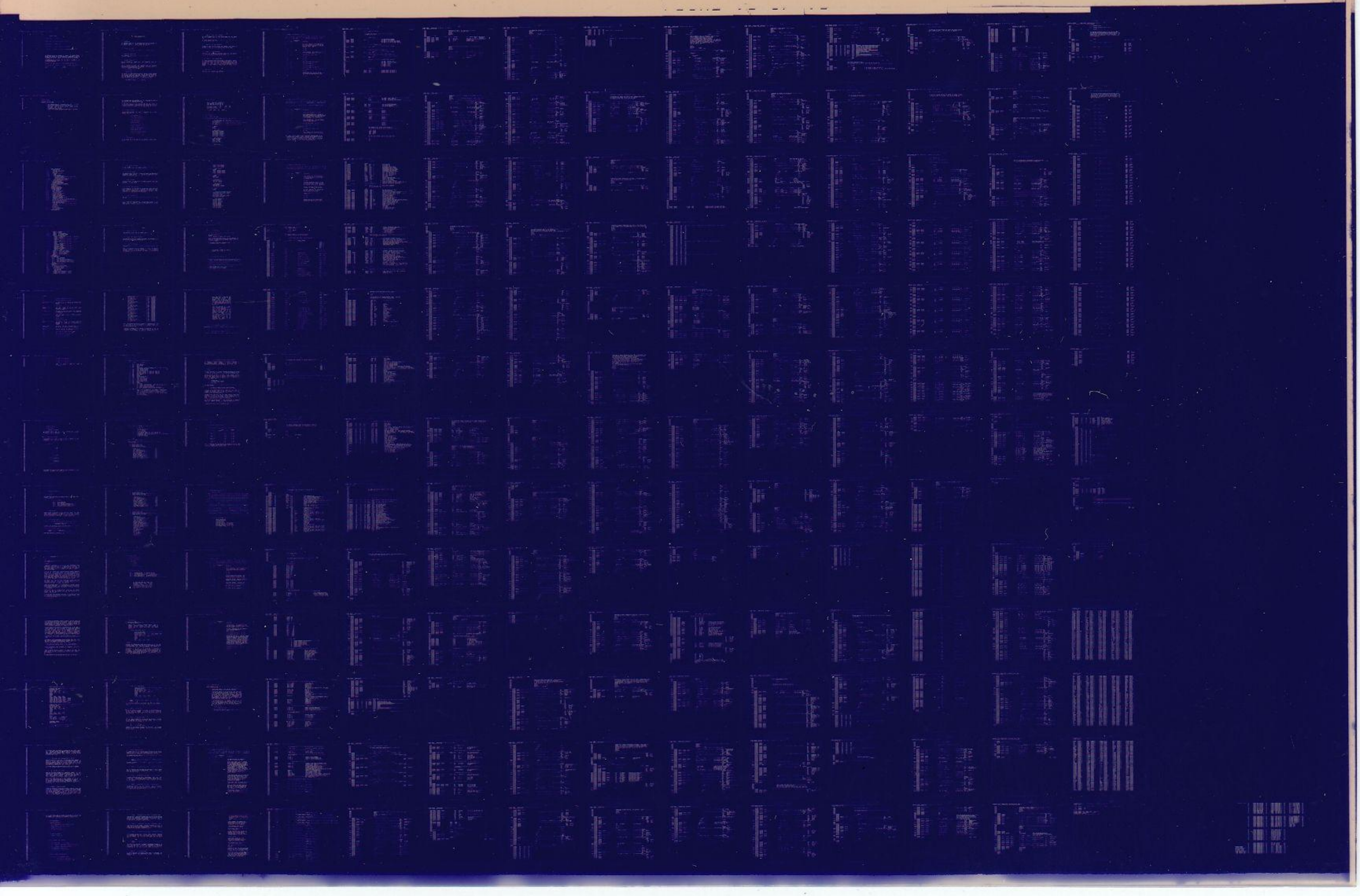
TK25

TK25 DATA RELIAB TST
CZTKIB0

AH-T784B-MC
1 OF 1 JUL 1985

MADE IN USA



```
                 .REM &

                          IDENTIFICATION
                          --------------

                      PRODUCT CODE:   AC-T783B-MC

                      PRODUCT NAME:   CZTKIBO TK25 DATA RELIABILITY TEST

                      PRODUCT DATE:   15-MARCH-1985

                      MAINTAINER:     MAGTAPE DIAGNOSTIC ENGINEERING

                      AUTHOR:         ROBERT F. WERY/JACK RICHARDSON/TERRENCE REILLY





                      THE INFORMATION IN THIS DOCUMENT IS  SUBJECT  TO  CHANGE WITHOUT
                      NOTICE  AND  SHOULD  NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
                      EQUIPMENT CORPORATION.  DIGITAL EQUIPMENT CORPORATION ASSUMES NO
                      RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

                      NO RESPONSIBILITY IS ASSUMED  FOR  THE  USE  OR  RELIABILITY  OF
                      SOFTWARE ON EQUIPMENT THAT IS NOT  SUPPLIED  BY  DIGITAL  OR ITS
                      AFFILIATED COMPANIES.

                      COPYRIGHT (C) 1984, 1985 BY DIGITAL EQUIPMENT CORPORATION

                      THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

                      DIGITAL        PDP            UNIBUS         MASSBUS
                      DEC            DECUS          DECTAPE
```

```
                        REVISION HISTORY:

                        REVISION B (25-OCT-1984) BY TERRENCE W. REILLY
                        CORRECTED THE FOLLOWING:

                              1)  DATA COMPARE ERROR ON A SUBSEQUENT UNIT, AFTER A DIFFERENT
                                  UNIT HAS BEEN DROPPED DUE TO A FATAL ERROR.
                              2)  WRITE RETRY ALGORITHM HAS BEEN CHANGED TO PERMIT 256 BAD
                                  TAPE SPOTS.  ONLY THE FIRST 20 WILL BE "MAPPED", BUT THE
                                  REMAINDER WILL BE TALLIED.
                              3)  A LINE CLOCK SERVICE ROUTINE HAS BEEN ADDED TO ELIMINATE
                                  SOME OF THE MACHINE DEPENDENT TIMING LOOPS.  (FOR THE 11/73)
```

```
                                USER'S GUIDE


1.0     GENERAL INFORMATION
1.1       PROGRAM ABSTRACT
1.1.1       FUNCTIONAL DESCRIPTION
1.1.2       STRUCTURE OF PROGRAM
1.1.3       MEMORY MAP
1.1.4       DIAGNOSTIC INFORMATION
1.1.4.1       SCOPE
1.1.4.2       ERROR RECOVERY
1.1.4.3       WRITE ERROR RECOVERY
1.1.4.3.1       MEDIA/OPERATIONAL SELECTIVE WRITE ERROR
                RECOVERY ALGORITHM
1.1.4.3.2       OPERATIONAL WRITE-ERROR-RECOVERY ALGORITHM
1.1.4.4       EARLY WARNING WRITE ERRORS
1.1.4.5       DIAGNOSTIC TIMING ADJUSTMENT
1.2       SYSTEM REQUIREMENTS
1.2.1       HARDWARE REQUIREMENTS
1.2.2       SOFTWARE REQUIREMENTS
1.3       RELATED DOCUMENTS AND STANDARDS
1.4       DIAGNOSTIC HIERARCHY PREREQUISITES
1.5       ASSUMPTIONS
1.6       DIAGNOSTIC HISTORY
2.0     OPERATING INSTRUCTIONS
2.1       HARDWARE PARAMETERS
2.2       SOFTWARE PARAMETERS
2.2.1       CLEAR COUNTERS
2.2.2       RESET RANDOM VARIABLES
2.2.3       PRINT SOFT ERRORS
2.2.4       INHIBIT RECOVERY
2.2.5       BAD TAPE SPOT DETECTION
2.2.6       DISABLE INTERRUPTS
2.2.7       INHIBIT RFC ERROR REPORTS
2.2.8       CONTROLLER RAM DUMP
2.2.9       ENABLE EARLY WARNING MESSAGES
2.2.10      CHANGE CMD SEQUENCE
2.2.11      COMMAND LIST FOR USE IN SOFTWARE DIALOGUE.
2.2.12      DATA PATTERN LIST FOR USE IN SOFTWARE DIALOGUE
2.3       EXAMPLES OF SOFTWARE DIALOGUE
2.3.1       BASIC FUNCTION AND DATA RELIABILITY WITH ALL
            ERROR REPORTING ENABLED
2.3.2       TO SET UP A SCOPE LOOP FOR A FAILURE IN BASIC
            FUNCTIONS.
2.3.3       TO SET UP A SCOPE LOOP FOR A FAILURE IN DATA
            RELIABILITY
2.4       EXECUTION TIMES
2.4.1       SYSTEM CONFIGURATION
2.4.2       TEST EXECUTION TIMES
3.0     ERROR INFORMATION
3.1       ERROR REPORTING
```

```
                                3.1.1       ERROR 1 - COMMAND PACKET ADDRESS NOT ON A
                                            MODULO 4 BOUNDRY:
                                3.1.2       ERROR 2 - TK25 NOT READY:
                                3.1.3       ERROR 3 - NO RESPONSE ERROR:
                                3.1.4       ERROR 4 - NO INTERRUPT ERROR:
                                3.1.5       SPECIAL CONDITION ERRORS:
                                3.1.5.1       ERROR 5 - TERMINATION CLASS CODE 0, UNDEFINED
                                              SPECIAL CONDITION
                                3.1.5.2       ERROR 6 - TERMINATION CLASS CODE 1, ATTENTION
                                              CONDITION
                                3.1.5.3       ERROR 7 - TERMINATION CLASS CODE 2, TAPE
                                              STATUS ALERT
                                3.1.5.4       ERROR 8 - TERMINATION CLASS CODE 3, FUNCTION
                                              REJECT
                                3.1.5.5       ERROR 9 - TERMINATION CLASS CODE 4,
                                              RECOVERABLE ERROR
                                3.1.5.6       ERROR 10 - TERMINATION CLASS CODE 5,
                                              RECOVERABLE ERROR
                                3.1.5.7       ERROR 11 - TERMINATION CLASS CODE 6,
                                              UNRECOVERABLE ERROR
                                3.1.5.8       ERROR 12 - TERMINATION CLASS CODE 7, FATAL
                                              SUBSYSTEM ERROR
                                3.1.6       ERROR 13 - RFC NON-ZERO ERROR:
                                3.1.7       ERROR 14 - RETRY LIMIT EXCEEDED:
                                3.1.8       ERROR 15 - TOO MANY INTERRUPTS:
                                3.1.9       ERROR 16 - CAPSTAN RUNAWAY:
                                3.1.10      ERROR 17 - DATA COMPARE ERROR:
                                3.2       ERROR HALTS
                                4.0     PERFORMANCE REPORT
                                5.0     TEST SUMMARIES
                                  5.1       TEST 1 - BASIC FUNCTIONS.
                                  5.2       TEST 2 - DATA RELIABILITY.
                                  5.3       TEST 3 - WRITE AND READ STREAMING TEST.
                                  5.4       TEST 4 - WRITE COMPATABILITY/WRITE UTILITY.
                                  5.5       TEST 5 - READ COMPATABILITY/READ UTILITY.
                                  5.6       TEST 6 - EXECUTE OPERATOR SELECTED COMMAND
                                            SEQUENCE.
                                6.0     DEVICE INFORMATION TABLES
                                  6.1       GENERAL
                                  6.2       BUS INTERFACE SPECIFICATIONS
                                  6.3       BIT DEFINITIONS FOR TK25 REGISTERS
                                    6.3.1     TK25 REGISTER SUMMARY
                                    6.3.2     TK25 STATUS REGISTER (TSSR)
                                    6.3.2.1     TSSR READ ONLY
                                    6.3.2.2     TSSR WRITE ONLY
                                    6.3.3     EXTENDED STATUS REGISTER 0 (XSTAT0)
                                    6.3.4     EXTENDED STATUS REGISTER 1 (XSTAT1)
                                    6.3.5     EXTENDED STATUS REGISTER 2 (XSTAT2)
                                    6.3.6     EXTENDED STATUS REGISTER 3 (XSTAT3)
```

GLOSSARY

ACT	AUTOMATED COMPUTER TEST SYSTEM
APT	AUTOMATED PRODUCT TEST SYSTEM
BYTE/RECORD/FILE COUNT BRF	IS STORED IN THE 4TH WORD OF THE COMMAND PACKET AND IT'S USE BY THE TK25 DEPENDS ON THE TYPE OF COMMAND.
CMD	TK25 COMMAND
COMMAND PACKET CMDPKT	FOUR WORD PACKET IN THE CPU MEMORY WHICH CONTAINS ALL INFORMATION NEEDED BY THE TK25 TO EXECUTE A COMMAND.
EXTENDED STATUS	FOUR WORDS OF TK25 STATUS WHICH ARE TRANSFERRED AS PART OF THE MESSAGE PACKET AT THE COMPLETION OF A COMMAND.
MESSAGE PACKET	SEVEN WORD PACKET IN THE CPU MEMORY INTO WHICH THE TK25 STORES STATUS AT THE COMPLETION OF A COMMAND.
PC	PROGRAM COUNTER
PSW	PROCESSOR STATUS WORD
RESIDUAL FRAME COUNT RFC	THIS COUNT IS PART OF THE MESSAGE PACKET AND CONTAINS THE NUMBER OF BYTES/RECORDS/FILES REMAINING TO BE PROCESSED AT THE COMPLETION OF A COMMAND.
SPECIAL CONDITION SPEC COND	TSS4 BIT15. WHEN SET, INDICATES THAT THE LAST COMMAND DID NOT COMPLETE WITHOUT INCIDENT.
TERMINATION CLASS CODE TCC	THREE BIT CODE IN THE TSSR WHICH INDICATES THE THE TYPE OF COMMAND TERMINATION.
TSBA	TAPE SYSTEM BUS ADDRESS REGISTER.
TSDB	TAPE SYSTEM DATA BUFFER REGISTER.
TSSR	TAPE SYSTEM STATUS REGISTER.
XST0	EXTENDED STATUS REGISTER 0

```
220                     XST1            EXTENDED STATUS REGISTER 1
221
222                     XST2            EXTENDED STATUS REGISTER 2
223
224                     XST3            EXTENDED STATUS REGISTER 3
225
226                     XXDP+           XXDP+  IS A  "CATCH-ALL"  NAME FOR  A  GROUP  OF
227                                     PDP-11   DIAGNOSTIC    PACKAGES   AVAILABLE   ON
228                                     MULTIMEDIA.
229
```

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

1.1.1 FUNCTIONAL DESCRIPTION -

THIS PROGRAM CAN BE USED AS A BASIC FUNCTION TEST, A DATA RELIABILITY TEST, A COMPATABILITY TEST, OR TO EXECUTE A SEQUENCE OF OPERATOR SELECTED COMMANDS.

1.1.2 STRUCTURE OF PROGRAM -

THIS DIAGNOSTIC IS A SINGLE PROGRAM FROM THE STANDPOINT OF THE DIAGNOSTIC USER, BUT IT USES A CONTROL MODULE RELEASED INDEPENDENTLY AS A DIAGNOSTIC SUPERVISOR.

1.1.3 MEMORY MAP -

```
            ---------------
      0    !               !
           !    VECTOR     !
           !     AREA      !
           !               !
    400    !---------------!
           !               !
           !     STACK     !
           !               !
   2000    !---------------!
           !               !
           !  DIAGNOSTIC   !
           !               !
  32000    !---------------!
           !  FREE MEMORY  !
  41000    !---------------!
           !               !
           !  DIAGNOSTIC   !
           !  SUPERVISOR   !
           !               !
  72000    !---------------!
           !               !
           !     XXDP+     !
           !               !
           !               !
  77777     ---------------
```

FREE MEMORY SPACE FOR THE WRITE/READ BUFFERS IS ALLOCATED BY THE SUPERVISOR ON REQUEST OR CHOSEN BY THE PROGRAMMER TO RESIDE BETWEEN THE DIAGNOSTIC AND THE SUPERVISOR.

1.1.4 DIAGNOSTIC INFORMATION -

1.1.4.1 SCOPE -

THIS DIAGNOSTIC CAN TEST UP TO FOUR (4) UNITS IN A ROUND ROBIN FASHION. THE FOUR UNITS ARE ASSIGNED LOGICAL UNIT NUMBERS 0 - 3 BY THE DIAGNOSTIC.

THERE ARE 6 TESTS IN THIS PROGRAM:

TEST 1 - BASIC FUNCTIONS.
TEST 2 - DATA RELIABILITY.
TEST 3 - WRITE AND READ STREAMING TEST.
TEST 4 - WRITE COMPATABILITY/WRITE UTILITY.
TEST 5 - READ COMPATABILITY/READ UTILITY.
TEST 6 - OPERATOR SELECTED SEQUENCE UTILITY.

1.1.4.2 ERROR RECOVERY -

ERROR RECOVERY IS PERFORMED ON READ, WRITE AND WRITE TAPE MARK ERRORS UNLESS RECOVERY IS INHIBITED BY THE OPERATOR. THE READ FORWARD/READ REVERSE RETRY LIMIT IS 16 (8 IN THE SAME DIRECTION AND 8 IN THE OPPOSITE DIRECTION). FOR MORE INFORMATION ON ERROR RECOVERY PROCEDURES, SEE SECTION 3.0 (ERROR INFORMATION).

1.1.4.3 WRITE ERROR RECOVERY -

THERE ARE 2 DISTINCT, SELECTABLE WRITE-ERROR-RECOVERY ALGORITHMS:

1. MEDIA/OPERATIONAL SELECTIVE ALGORITHM
2. OPERATIONAL ALGORITHM

BY DEFAULT THE DIAGNOSTIC SELECTS THE FIRST ALGORITHM TO DISCERN MEDIA RELATED WRITE ERRORS FROM OPERATIONAL ONES.

TO SELECT THE SECOND ALGORITHM:

ANSWER 'Y' TO CHANGE SW (L) ?
ANSWER 'N' TO BAD TAPE SPOT DETECTION (L) Y ?

WHEN ERROR RECOVERY IS INHIBITED, THE LATTER QUESTION IS NOT ASKED AND BOTH ALGORITHMS ARE BYPASSED.

1.1.4.3.1 MEDIA/OPERATIONAL SELECTIVE WRITE ERROR RECOVERY ALGORITHM -

SCOPE

THE ALGORITHM DISCERNS MEDIA RELATED WRITE ERRORS FROM OPERATIONAL ONES.

ALGORITHM

A WRITE RETRY SUBROUTINE IS CALLED BY THE RECOVERABLE ERROR SUBROUTINE UPON DETECTION OF A RECOVERABLE WRITE ERROR. THE WRITE RETRY SUBROUTINE REWRITES THE RECORD IN THE SAME SPOT ON TAPE FOUR TIMES. IF ALL 4 REPEATS ARE GOOD, THE RECORD IS CONSIDERED RECOVERED AND A RECOVERABLE WRITE ERROR IS LOGGED AT THAT RECORD NUMBER.

IF ANY OF THE 4 REPEATS FAIL, THE BAD RECORD IS ERASED, SUSPECTED BAD SPOT AT THAT RECORD NUMBER IS LOGGED, AND THE RECORD IS RETRIED AGAIN 3 INCHES FURTHER DOWN TAPE. THIS IS DONE UP TO 4 TIMES, UP TO 4 REPEATS EACH. IF THE RECORD CANNOT BE WRITTEN WITHOUT RECOVERABLE ERRORS AFTER 4 RETRIES, THE RECORD IS ERASED AND A BAD SPOT DETECTED ON RETRY FAILURE IS REPORTED. THE RECOVERABLE ERROR SUBROUTINE THEN CONTINUES TO CALL THE WRITE RETRY SUBROUTINE, WHICH REISSUES THE GROUP OF 4 RETRIES, UNTIL THE RECORD IS RECOVERED OR 20 BAD SPOTS HAVE BEEN LOGGED.

TWO HUNDRED FIFTY (250) BAD SPOTS MAXIMUM ARE ALLOWED PER TAPE PER PASS. WHEN 250 BAD SPOTS HAVE BEEN LOGGED, THE TAPE IS CONSIDERED DEFECTIVE. A BAD TAPE OVERFLOW MESSAGE WILL BE PRINTED AND THE UNIT IS REWOUND, THEN DROPPED.

DURING THE RECOVERY PROCESS, IT IS NECESSARY TO PERFORM SEVERAL TAPE POSITION OPERATIONS. IF A POSITION ERROR STATUS IS DETECTED DURING THOSE OPERATIONS, THEN THE RECOVERY ATTEMPT IS ABORTED AND AN APPROPRIATE UNRECOVERABLE ERROR MESSAGE IS PRINTED. THE UNIT IS THEN DROPPED.

RECORDS WHICH WERE NOT WRITTEN WITHOUT ERROR WILL BE ERASED. THIS IS SO THAT ALL RECORDS LEFT ON TAPE ARE GOOD WRITTEN RECORDS. BAD SPOTS ARE ERASED, WITH ERASE GAPS FROM 3 TO 12 INCHES PER RETRY GROUP.

IF NO BAD SPOTS WERE PREVIOUSLY DETECTED, UP TO 20 FEET OF ERASE GAP COULD RESULT WHEN RETRYING TO RECOVER A SINGLE RECORD. THAT LONG STRETCH OF BAD TAPE WOULD THEN BE FLAGGED WITH 20 BAD SPOTS AT THE SAME RECORD NUMBER.

BAD SPOTS REPORTS

IF THE PRINT OF RECOVERABLE ERRORS IS ENABLED, THE BAD SPOTS ON TAPE ARE IDENTIFIED AS THEY ARE DETECTED. SINCE THE BAD RECORDS ARE ERASED UNTIL RECOVERED, THE BAD SPOTS ACTUALLY PRECEDE THE RECORD NUMBER THAT IDENTIFIES THEM. THE NUMBER OF REPEATS AND RETRIES ATTEMPTED IS PRINTED, FROM WHICH THE LENGTH OF ERASE GAPS CAN BE DETERMINED (APPROXIMATELY 3 INCHES PER RETRY).

THE STATISTICAL REPORT PRINTED AT THE END OF TEST 2 OR UPON A "PRINT" REQUEST, CONTAINS A SUMMARY OF THE BAD SPOTS LOGGED ON THE CURRENT TAPE PASS. IN THAT REPORT, ALL COUNTS ARE CUMULATIVE FROM PASS TO PASS, EXCEPT FOR THE NUMBER OF BAD SPOTS: IT RELATES TO A "TAPE PASS" ONLY. FOR THIS PURPOSE, A "TAPE PASS" IS A WRITE PASS FROM BOT TO EOT, OR FROM BOT TO WHERE THE DIAGNOSTIC IS HALTED BEFORE REACHING EOT. A PASS IS DEFINED BY THE SUPERVISOR AS A RUN THROUGH ALL THE TESTS REQUESTED ON ALL UNITS SELECTED. THESE PASSES ARE IDENTIFIED AS "PASS" AND "EOP".

THE NUMBER OF WRITE RETRIES, CUMULATIVE FROM PASS TO PASS, IS A GLOBAL COUNT OF HOW MANY TIMES THE GROUP OF 4 RETRIES HAS BEEN CALLED.

THE NUMBER OF WRITE RECOVERABLE ERRORS EXCLUDES BAD TAPE SPOTS AND REFLECTS THE SPECIFICATIONS OF THE HARDWARE UNDER TEST. PER TAPE PASS, THE NUMBER OF WRITE RETRIES EQUALS THE SUM OF THE NUMBER OF RECOVERABLE WRITE ERRORS AND BAD SPOTS.

TO CLEAR CUMULATIVE COUNTS, ANSWER 'Y' TO: CLEAR COUNTERS (L) Y ?. THE BAD TAPE SPOTS COUNT IS CLEARED WHEN WRITING FROM BOT.

IF TEST 2 IS HALTED, THEN RESTARTED OR CONTINUED, THE RECORD COUNT IS RESET TO ZERO AND THE BAD SPOT ID SHALL FOLLOW THAT
    RESET COUNT.

SINCE ALL WRITTEN RECORDS ULTIMATELY ARE KNOWN TO BE GOOD, THE READ ERRORS CAN BE ATTRIBUTED TO TRANSIENT NOISE, TRANSIENT ELECTRICAL MALFUNCTIONS, OR CONTAMINANTS ON TAPE AS OPPOSED TO TAPE DEFECTS.

THE SAME RECORDS MUST BE WRITTEN FROM TAPE PASS TO TAPE PASS FOR THE BAD SPOTS ID TO REMAIN CONSISTENT IN THOSE TAPE PASSES.

```
                        EXAMPLE OF A TAPE PASS PRINT:

                        CZTKIA SFT ERR  00009 ON UNIT 00 TST 002 SUB 000 PC: 012100
                        RECOVERABLE ERROR
                        WRT CMD FAILED - UNIT 0   PASS:    1   RECORD:    6
                        PREVIOUS CMD WAS  WRT
                        CMDPKT  TSBA    RFC     TSSR   TCC
                        100205  002406  000000  100210  4
                        026600
                        000000
                        003107
                        XST0    XST1    XST2    XST3
                        000350  000002  100400  000000
                        SUSPECT BAD SPOT AFTER 1 RETRY, 2 REPEAT
                        SUSPECT BAD SPOT AFTER 2 RETRY, 1 REPEAT
                        SUSPECT BAD SPOT AFTER 3 RETRY, 1 REPEAT
                        SUSPECT BAD SPOT AFTER 4 RETRY, 3 REPEAT
                        RETRY FAILED ON BAD SPOT...ERASED!
                        SUSPECT BAD SPOT AFTER 1 RETRY, 1 REPEAT

                        CZTKIA SFT ERR  00009 ON UNIT 00 TST 002 SUB 000 PC: 012100
                        RECOVERABLE ERROR
                        WRT CMD FAILED - UNIT 0   PASS:    1   RECORD:10210
                        PREVIOUS CMD WAS  WRT
                        CMDPKT  TSBA    RFC     TSSR   TCC
                        100205  002406  000000  100210  4
                        026600
                        000000
                        004000
                        XST0    XST1    XST2    XST3
                        000350  000002  100010  000000
                        RECOVERED ON RETRY # 1
                        ↑C
                        DR>PRI
                        UNIT 0   PASS:    1   RECORD:10210
                        BYTES WRITTEN    0,272,279,691
                        BYTES READ REV   0,301,123,654
                        BYTES READ REV   0,301,120,381
                                              WRT    RDR    RDF
                        RECOVERABLE ERRORS     1      0      0
                        UNRECOVERABLE ERRORS   0      0      0
                        WRITE RETRIES          3

                         2 BAD SPOTS THIS TAPE PASS PRECEDING RECORD #:
                                  6      6
                        SPEC COND   HARD   FATAL   COMPARE
                              2      0       0       0
                        DR>
```

THIS EXAMPLE SHOWS THAT RECORD 6 RECOVERED ON 2ND RETRY GROUP. THE 2 BAD SPOTS RESIDE IN A 18 INCH ERASE GAP BETWEEN RECORDS 5 AND 6. RECORD 10210 RECOVERED ON 1ST RETRY OF 4 GOOD REPEATS. THREE WRITE GROUP RETRIES WERE ATTEMPTED, RESULTING IN ONE RECOVERABLE WRITE ERROR FROM RECORD 10210 AND TWO BAD SPOTS BETWEEN RECORDS 5 AND 6.

1.1.4.3.2 OPERATIONAL WRITE-ERROR-RECOVERY ALGORITHM -

WHEN THIS ALGORITHM IS SELECTED, THE TK25 WRITE RETRY COMMAND IS ISSUED UP TO 16 TIMES OR UNTIL THE RECORD IS RECOVERED. THE WRITE RETRY COMMAND CONSISTS OF A SPACE REVERSE OVER THE BAD RECORD, AN ERASE OF 3 INCHES OF TAPE AND A REWRITE OF THE RECORD. THAT COMPOSITE COMMAND DOES NOT DETECT BAD SPOTS ON TAPE. THEREFORE NO BAD TAPE SPOTS STATUS IS PRINTED.

IF A RECORD CANNOT BE RECOVERED AFTER 16 WRITE RETRY COMMANDS, A RETRY LIMIT EXCEEDED ERROR IS FLAGGED AND THE UNIT IS DROPPED.

1.1.4.4 EARLY WARNING WRITE ERRORS -

SINCE THE TK25 DRIVE RECORDS IN A SERPENTINE MANNER, THE TAPE CARTRIDGE HAS PHYSICAL MARKERS AT EACH END OF THE TAPE. THE TK25 CONTROLLER WILL NOT ALLOW THE WRITING OF DATA OVER THE AREA OF THESE "EARLY WARNING" MARKERS. THEREFORE, WHEN AN ATTEMPT IS MADE TO WRITE DATA AT THE END OF TRACK, A WRITE ERROR STATUS (TCC4) IS RETURNED WITH THE EARLY WARNING (EW) BIT SET.

WHEN A WRITE ERROR OCCURS AND THE EARLY WARNING BIT IS SET IN XSTAT1, THE ERROR IS RETRIED IF ERROR RECOVERY IS ENABLED. THE ONLY METHOD OF RETRY USED FOR EARLY WARNING WRITE ERRORS IS TO SET THE RETRY MODIFIER ON THE ORIGINAL COMMAND AND TO REISSUE IT. IN ADDITION, EARLY WARNING WRITE ERRORS ARE NOT COUNTED IN ANY ERROR TALLIES.

1.1.4.5 DIAGNOSTIC TIMING ADJUSTMENT -

A NUMBER OF SUPERVISOR TIMING DELAYS MACROS, KNOWN AS WATCH DOG DELAYS, ARE CALLED BY THE DIAGNOSTIC TO WAIT FOR VARIOUS COMMANDS TO COMPLETE. THESE DELAYS ARE NOT CALIBRATED AND SIMPLY EXPAND INTO INLINE NESTED LOOPS. THE COUNT FOR THE OUTER LOOP COMES FROM THE VARIABLE ARGUMENT SUPPLIED BY THE DELAY CALLS. THE COUNT FOR THE INNER LOOP COMES FROM THE FIXED "HEADER" ELEMENT "L$DLY". AS THE DIAGNOSTIC IS RUN ON DIFFERENT CPU'S, THESE DELAYS WILL VARY IN LENGTH WITH MEMORY SPEED.

IF TIME-OUT OCCURS WHEN NO APPARENT MALFUNCTIONS IN THE TAPE UNIT ARE EVIDENT, THE TIMINGS OF THE DIAGNOSTIC MAY BE ADJUSTED. THIS IS ACCOMPLISHED BY PATCHING THE FIXED DELAY ELEMENT "L$DLY".

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE REQUIREMENTS -

- 0 PDP-11 PROCESSOR WITH 16K OR MORE OF MEMORY
- 0 CONSOLE DEVICE (LA36,LA120,VT100,ETC.)
- 0 PROGRAM LOAD DEVICE
- 0 TK25 DRIVE AND CONTROLLER

1.2.2 SOFTWARE REQUIREMENTS -

- 0 DIAGNOSTIC SUPERVISOR

1.3 RELATED DOCUMENTS AND STANDARDS

- 0 XXDP+ USERS MANUAL MD-11-CHQUS
- 0 PDP-11 DIAGNOSTIC SUPERVISOR INTERFACE SPECIFICATION
- 0 PDP-11 DIAGNOSTIC SUPERVISOR PROGRAMMER'S GUIDE
- 0 TK25 PROGRAMMING SPECIFICATION
- 0 TK25 COMMAND PACKET SPECIFICATION

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

ORDER OF HOST CPU DIAGNOSTIC USAGE:

1) FRONT END FUNCTIONAL PROGRAM - ALL TESTS.
2) DATA RELIABILITY PROGRAM:

A) BASIC FUNCTION TEST.
B) DATA RELIABILITY TEST.

1.5 ASSUMPTIONS

THE HARDWARE OTHER THAN THE SUBSYSTEM BEING TESTED IS ASSUMED TO BE WORKING PROPERLY. FALSE ERRORS MAY BE REPORTED IF THE PROCESSOR, MEMORY, ETC., DOES NOT FUNCTION PROPERLY.

1.6 DIAGNOSTIC HISTORY

REVISION A - 15-MAR-84 - ORIGINAL RELEASE

2.0 OPERATING INSTRUCTIONS

FOR OPERATING INSTRUCTIONS, PLEASE SEE CHAPTER 5 OF XXDP+ OPERATOR'S MANUAL.

2.1 HARDWARE PARAMETERS

ON A "N" RESPONSE TO "CHANGE HW?", THE DIAGNOSTIC SHALL RUN ASSUMING THAT THERE IS ONE UNIT AT TSSR = 172522 WITH A VECTOR = 224.

ON A "Y" RESPONSE TO "CHANGE HW?" QUESTION,THEN THE FOLLOWING QUESTIONS WILL BE ASKED ON A START COMMAND. THE VALUE LOCATED TO THE LEFT OF THE QUESTION MARK IS THE DEFAULT VALUE THAT WILL BE TAKEN ON A CARRIAGE RETURN RESPONCE.

```
TSSR ADDRESS (172522) ?

VECTOR (224) ?
```

THE VALIDITY OF THESE PARAMETERS CAN BE CHECKED BEFORE RUNNING THE TESTS BY SETTING THE FLAG "ADR" ON A STA, RES OR CON COMMAND. THE SO CALLED "AUTO DROP" CODE SHALL THEN BE EXECUTED AFTER THE INIT CODE AND BEFORE THE HARDWARE TESTS ARE RUN. THAT CODE FIRST TESTS THE ADDRESS OF THE TSSR(S). IF THERE IS NO RESPONSE, IT DROPS THE UNIT(S) IMMEDIATELY WITH THE FOLLOWING MESSAGE:

```
BUS TRAP AT XXXXXX          ( XXXXXX = TSSR AD )
INTERFACE BAD OR NOT SET TO ABOVE AD.
```

ON A RESPONSE FROM THE INTERFACE, THE UNITS THAT ARE NOT READY OR NOT ON-LINE ARE DROPPED IMMEDIATELY. THE HARDWARE TESTS SHALL THEN BE RUN ON THE RESPONDING UNITS.

IF THE "ADR" FLAG IS NOT SET, THE READY AND OFF-LINE STATUS OF THE UNITS ARE CHECKED. A MESSAGE SHALL BE PRINTED EVERY SO OFTEN TO WARN THE OPERATOR OF UNITS THAT ARE NOT READY OR OFF-LINE. THESE UNITS SHALL BE DROPPED AFTER A REASONABLE AMOUNT OF TIME.

2.2 SOFTWARE PARAMETERS

THE FOLLOWING QUESTIONS ARE ASKED IF REQUESTED ON A START, RESTART, OR CONTINUE. THEY ALLOW FLEXABILITY IN THE WAY THE PROGRAM BEHAVES.

CLEAR COUNTERS (L) Y ?

RESET RANDOM VARIABLES (L) N ?

HALT AFTER EACH CMD (L) N ?

PRINT SOFT ERRORS (L) N ?

INHIBIT RECOVERY (L) N ?

BAD TAPE SPOT DETECT (L) Y ?

DISABLE INTERRUPTS (L) N ?

INHIBIT RFC ERROR REPORTS (L) N ?

CONTROLLER RAM DUMP (L) N ?

ENABLE EARLY WARNING MESSAGES (L) N ?

CHANGE CMD SEQUENCE (L) N ?

2.2.1 CLEAR COUNTERS -

IF YOU ANSWER YES TO THIS QUESTION, ALL COUNTERS (ERROR COUNTS, SPECIAL CONDITION COUNT, BYTES TRANSFERRED, ETC.) WILL BE SET TO ZERO.

2.2.2 RESET RANDOM VARIABLES -

IF YOU REQUEST THAT THE RANDOM VARIABLES BE RESET, THESE VARIABLES WILL TAKE ON THEIR DEFAULT VALUES.

2.2.3 PRINT SOFT ERRORS -

THIS QUESTION WILL ALLOW YOU TO ENABLE OR DISABLE THE PRINTING OF RECOVERABLE ERROR REPORTS. IF YOU ARE ONLY INTERESTED IN THE SUMMARY OF ERRORS AND NOT THE INDIVIDUAL ERRORS, YOU SHOULD ANSWER (N) TO THIS QUESTION.

2.2.4 INHIBIT RECOVERY -

IF YOU SO CHOOSE, YOU CAN INHIBIT ALL ERROR RECOVERY WITH THIS QUESTION. IF YOU ANSWER (Y) TO THIS QUESTION, THE FOLLOWING QUESTION WILL NOT BE ASKED.

2.2.5 BAD TAPE SPOT DETECTION -

IF YOU ANSWER (Y) TO THIS QUESTION, THE BAD TAPE SPOT DETECTION RETRY ALGORITHM WILL BE USED. OTHERWISE, THE OPERATIONAL WRITE ERROR RECOVERY ALGORITHM WILL BE USED FOR ERROR RECOVERY. THESE ALGORITHMS ARE DESCRIBED IN SECTIONS ABOVE. THIS QUESTION WILL NOT BE ASKED IF ERROR RECOVERY IS INHIBITED.

2.2.6 DISABLE INTERRUPTS -

YOU CAN DISABLE OR ENABLE CONTROLLER INTERRUPTS WITH THIS QUESTION.

2.2.7 INHIBIT RFC ERROR REPORTS - THIS QUESTION ALLOWS YOU TO INHIBIT THE RFC (RESIDUAL FRAME COUNT) ERROR REPORTS. THESE ERROR REPORTS INDICATE THAT N BYTES/RECORDS/FILES WERE NOT PROCESSED AT THE END OF THE COMMAND.

757
758
759
760 2.2.8 CONTROLLER RAM DUMP -
761
762 THIS QUESTION ALLOWS YOU TO CHOOSE WHETHER A DUMP OF THE
763 CONTOLLER'S RAM WILL OCCUR WITH EACH ERROR.
764
765
766
767 2.2.9 ENABLE EARLY WARNING MESSAGES -
768
769 THIS QUESTION ALLOWS YOU TO ENABLE OR DISABLE THE PRINTING OF A
770 MESSAGE EACH TIME A WRITE ERROR OCCURS AND THE EARLY WARNING (EW)
771 BIT IS SET IN EXTENDED STATUS REGISTER 1.
772
773
774
775 2.2.10 CHANGE CMD SEQUENCE -
776
777 THIS QUESTION WILL ALLOW YOU TO CHANGE THE COMMAND SEQUENCE USED
778 IN TEST 6.
779
780 NOTE: THIS QUESTION SHOULD BE ANSWERED (N) UNLESS AN OPERATOR
781 SELECTED SEQUENCE IS TO BE EXECUTED. IF THIS QUESTION IS
782 ANSWERED (N), NO MORE QUESTIONS WILL BE ASKED. IF THIS QUESTION
783 IS ANSWERED (Y), THE FOLLOWING QUESTIONS MUST BE ANSWERED OR
784 DEFAULTED WITH A CR>:

```
                        CHARACTERISTICS CODE (O) 40 ?   (0,20,40,200) (OCTAL)
                        CMD/2 (D) 13 ?                  (1-27)    (DECIMAL)
                        BRF COUNT (D) 1 ?               (1-4K)    (DECIMAL)
                        # OF OPERATIONS (D) 1 ?         (1-32K)   (DECIMAL)
                        PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
                        CMD/3 (D) 4 ?                   (1-27)    (DECIMAL)
                        BRF COUNT (D) 4096 ?            (1-4K)    (DECIMAL)
                        # OF OPERATIONS (D) 11719 ?     (1-32K)   (DECIMAL)
                        PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
                        CMD/4 (D) 7 ?                   (1-27)    (DECIMAL)
                        BRF COUNT (D) 4096 ?            (1-4K)    (DECIMAL)
                        # OF OPERATIONS (D) 11719 ?     (1-32K)   (DECIMAL)
                        PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
                        CMD/5 (D) 2 ?                   (1-27)    (DECIMAL)
                        BRF COUNT (D) 4096 ?            (1-4K)    (DECIMAL)
                        # OF OPERATIONS (D) 11719 ?     (1-32K)   (DECIMAL)
                        PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
                        CMD/6 (D) 13. ?                 (1-27)    (DECIMAL)
                        BRF COUNT (D) 1 ?               (1-4K)    (DECIMAL)
                        # OF OPERATIONS (D) 1 ?         (1-32K)   (DECIMAL)
                        PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
                        CMD/7 (D) 27. ?                 (1-27)    (DECIMAL)
                        BRF COUNT (D) 4096 ?            (1-4K)    (DECIMAL)
                        # OF OPERATIONS (D) 11719 ?     (1-32K)   (DECIMAL)
                        PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
                        CMD/8 (D) 27. ?                 (1-27)    (DECIMAL)
                        BRF COUNT (D) 4096 ?            (1-4K)    (DECIMAL)
                        # OF OPERATIONS (D) 11719 ?     (1-32K)   (DECIMAL)
                        PATTERN (D) 7 ?                 (0-8)     (DECIMAL)
```

NOTE: THE PROGRAM AUTOMATICALLY INSERTS A CHARACTERISTICS CODE 40 (SEE BELOW) AS THE FIRST COMMAND IN THE SEQUENCE TABLE. IF A DIFFERENT CHARACTERISTIC IS DESIRED, THE OPERATOR SHOULD ENTER THAT CHARACTERISTIC CODE.

A TOTAL OF 7 COMMANDS MAY BE ENTERED IN ADDITION TO THE SET CHARACTERISTICS COMMAND. IF THE OPERATOR WISHES TO USE LESS THAN 7 COMMANDS, AN END COMMAND (27) MUST BE ENTERED AND THEN A CONTROL Z (Z) CAN BE ENTERED TO TERMINATE SOFTWARE DIALOGUE.

831
832
833
834 2.2.11 COMMAND LIST FOR USE IN SOFTWARE DIALOGUE. -
835
836 CODE COMMAND DESCRIPTION
837
838 1 = DRI DRIVE INITIATE.
839 2 = RDF READ FORWARD.
840 3 = RDR READ REVERSE.
841 4 = WRT WRITE.
842 5 = WTV WRITE/VERIFY. (WRITE N RECORDS; SPACE REVERSE N RECORDS;
843 READ FORWARD AND CHECK N RECORDS.)
844 6 = SRF SPACE RECORDS FORWARD.
845 7 = SRR SPACE RECORDS REVERSE.
846 8 = RNR READ NEXT REVERSE, IE. SPACE FWD, READ REV.
847 9 = RNF READ NEXT FORWARD, IE. READ FWD, SPACE REV.
848 10 = RPF READ PREVIOUS FWD, IE. SPACE REV, READ FWD.
849 11 = RPR READ PREVIOUS REV, IE. READ REV, SPACE FWD.
850 12 = WRR WRITE RETRY.
851 13 = RWD REWIND.
852 14 = MBR MESSAGE BUFFER RELEASE.
853 15 = WTM WRITE TAPE MARK.
854 16 = WTR WRITE TAPE MARK RETRY.
855 17 = SFF SPACE FILES FORWARD.
856 18 = SFR SPACE FILES REVERSE.
857 19 = GES GET EXTENDED STATUS.
858 20 = ERS ERASE 3 INCHES OF TAPE.
859 21 = UNL UNLOAD.
860 22 = CLN CLEAN TAPE
861 23 = SCH SET DEVICE CHARACTERISTIC. WHERE BRF=200, 40, 20, 0.
862 200 = ENABLE SKIP TAPE MARKS STOP (STOP AT LOGICAL EOT)
863 40 = ENABLE ATTENTION INTERRUPTS.
864 20 = ENABLE MESSAGE BUFFER RELEASE INTERRUPTS.
865 SEE TK25 PROGRAMMING SPECIFICATION FOR DESCRIPTION.
866 24 = NOT USED
867 25 = JMP JUMP TO THE NTH COMMAND IN THE COMMAND SEQUENCE TABLE,
868 WHERE N IS DEFINED IN THE BRF FIELD. THE NUMBER OF
869 JUMPS IS ENTERED IN THE NUMBER OF OPERATIONS FIELD.
870 26 = DLY DELAY "N" MILISECONDS WHERE N IS DEFINED IN THE NUMBER
871 OF OPERATIONS. THIS DELAY IS USED BETWEEN EACH
872 EXECUTABLE COMMAND.
873 27 = END END OF COMMAND SEQUENCE.

875
876
877
878 2.2.12 DATA PATTERN LIST FOR USE IN SOFTWARE DIALOGUE -
879
880 PATTERN # DESCRIPTION.
881
882 0 INCREMENTING PATTERN. 0 - 377.
883 1 ALL 1'S PATTERN.
884 2 ALL 0'S PATTERN.
885 3 1 BIT WALKING FROM R TO L IN A FIELD OF 0'S.
886 4 0 BIT WALKING FROM R TO L IF A FIELD OF 1'S.
887 5 ALTERNATING 1 AND 0 BITS WITH ALTERNATE BYTES COMPLIMENTED. (125/25
888 6 ALTERNATING BYTES OF 000 AND 377.
889 7 RANDOM DATA PATTERN.
890 8 NO PATTERN GENERATION.
891
892
893
894 2.3 EXAMPLES OF SOFTWARE DIALOGUE
895
896 2.3.1 BASIC FUNCTION AND DATA RELIABILITY WITH ALL ERROR
897 REPORTING ENABLED -
898
899 A) RECEIVE PROMPT (DR>)
900 B) ENTER STA/TES:1-2<CR>
901 C) ANSWER HARDWARE QUESTIONS.
902 D) PROCEED WITH THE FOLLOWING DIALOGUE:
903
904 CHANGE SW (L) ? Y<CR>
905 CLEAR COUNTERS (L) N ? Y<CR>
906 RESET RANDOM VARIABLES (L) N ? N<CR>
907 HALT AFTER EACH CMD (L) N ? N<CR>
908 PRINT SOFT ERRORS (L) N ? Y<CR>
909 INHIBIT RECOVERY (L) N ? N<CR>
910 BAD TAPE SPOT DETECT (L) Y ? Y<CR>
911 DISABLE INTERRUPTS (L) N ? N<CR>
912 INHIBIT RFC ERROR REPORT (L) N ? N<CR>
913 CONTROLLER RAM DUMP (L) N ? N<CR>
914 ENABLE EARLY WARNING MESSAGES (L) N ? N<CR>
915 CHANGE CMD SEQUENCE (L) N ? N<CR>

2.3.2 TO SET UP A SCOPE LOOP FOR A FAILURE IN BASIC FUNCTIONS. -

A) RECEIVE PROMPT (DR>)
B) ENTER STA/TES:1/FLA:LOE:IER:ISR:IDU<CR>
C) ANSWER HARDWARE QUESTIONS.
D) PROCEED WITH THE FOLLOWING DIALOGUE:

```
CHANGE SW (L) ?                          Y<CR>
CLEAR COUNTERS (L) N ?                   Y<CR>
RESET RANDOM VARIABLES (L) N ?           N<CR>
HALT AFTER EACH CMD (L) N ?              N<CR>
PRINT SOFT ERRORS  (L) N ?               N<CR>
INHIBIT RECOVERY (L) N ?                 N<CR>
BAD TAPE SPOT DETECT (L) Y ?             N<CR>
DISABLE INTERRUPTS (L) N ?               N<CR>
INHIBIT RFC ERROR REPORT (L) N ?         Y<CR>
CONTROLLER RAM DUMP (L) N ?              N<CR>
ENABLE EARLY WARNING MESSAGES (L) N ?    N<CR>
CHANGE CMD SEQUENCE (L) N ?              N<CR>
```

2.3.3 TO SET UP A SCOPE LOOP FOR A FAILURE IN DATA RELIABILITY -

A) RECEIVE PROMPT (DR>)
B) ENTER STA/TES:5/FLA:IER:ISR:IDU/EOP:1000<CR>
C) ANSWER HARDWARE QUESTIONS.
D) PROCEED WITH THE FOLLOWING DIALOGUE:

```
CHANGE SW (L) ?                          Y<CR>
CLEAR COUNTERS (L) N ?                   Y<CR>
RESET RANDOM VARIABLES (L) N ?           N<CR>
HALT AFTER EACH CMD (L) N ?              N<CR>
PRINT SOFT ERRORS  (L) N ?               N<CR>
INHIBIT RECOVERY (L) N ?                 N<CR>
BAD TAPE SPOT DETECT (L) Y ?             N<CR>
DISABLE INTERRUPTS (L) N ?               Y<CR>
INHIBIT RFC ERROR REPORT (L) N ?         Y<CR>
CONTROLLER RAM DUMP (L) N ?              N<CR>
ENABLE EARLY WARNING MESSAGES (L) N ?    N<CR>
CHANGE CMD SEQUENCE (L) N ?              Y<CR>
CHARACTERISTICS CODE (O) 40 ?           40<CR>
CMD/2 (D) 5 ?                           13<CR>   (REWIND) (COULD BE ANY COMMA
BRF COUNT (D) 4096 ?                     1<CR>
# OF OPERATIONS (D) 10 ?                1<CR>
PATTERN (D) 7 ?                          1<CR>
CMD/3 (D) 5 ?                            4<CR>   (WRITE) (COULD BE ANY COMMAN
BRF (D) 4096 ?                        1000<CR>
# OF OPERATIONS (D) 10 ?              10000<CR>
PATTERN (D) 7 ?                          1<CR>
CMD/4 (D) 5 ?                           27<CR>   (END)   (COULD BE ANY COMMAN
BRF (D) 4096 ?                           <^Z>
```

2.4 EXECUTION TIMES

2.4.1 SYSTEM CONFIGURATION -

- 0 PDP-11/23+ PROCESSOR
- 0 128KW MEMORY
- 0 LA34 TERMINAL
- 0 1 - TK25 DRIVE
- 0 1 - TK25 Q-BUS CONTROLLER

2.4.2 TEST EXECUTION TIMES -

TEST 1 - BASIC FUNCTIONS - 0.5 MINUTES PER PASS.
TEST 2 - DATA RELIABILITY - 83 MINUTES PER PASS.
TEST 3 - WRITE/READ STREAMING TEST - 53 MINUTES PER PASS.
TEST 4 - WRITE COMPATABILITY - 34 MINUTES PER PASS.
TEST 5 - READ COMPATABILITY - 23 MINUTES PER PASS.
TEST 6 - OPERATOR SELECTED SEQUENCE - 48 MINUTES (FOR DEFAULT SEQUENCE).

NOTE

ALL EXECUTION TIMES ARE SHOWN FOR ONE UNIT OPERATION USING A 600 FOOT CARTRIDGE AND THE SYSTEM CONFIGURATION ABOVE. FOR SYSTEM CONFIGURATIONS OTHER THAN AN 11/23+ WITH 128KW OF MEMORY AND A SINGLE TK25 SUBSYSTEM, TEST DURATIONS WILL VARY.

3.0 ERROR INFORMATION

3.1 ERROR REPORTING

ALL ERROR REPORTS EXCEPT FOR ERRORS 1 AND 17 INCLUDE A DUMP OF THE FOLLOWING INFORMATION:

ERROR #, TEST #, SUBTEST #, PROGRAM COUNTER, UNIT #, COMMAND, PREVIOUS COMMAND, PASS COUNT, NUMBER OF RECORDS FROM BOT, RECORD READ COUNT, THE COMMAND PACKET, TSSR, TCC, TSBA, RFC, AND THE EXTENDED STATUS REGISTERS.

STANDARD ERROR REPORT FORMAT:

```
CZTKIA SFT ERR XXXXX  TST XXX  SUB XXX  PC: XXXXXX
(ASCII ERROR MESSAGE)
XXX CMD FAILED - UNIT X  PASS: XXXXX  RECORD: XXXXX
PREVIOUS CMD WAS  XXX          * RECORD READ: XXXXX *
CMDPKT  TSBA    RFC     TSSR    TCC
XXXXXX  XXXXXX  XXXXXX  XXXXXX  X
XXXXXX
XXXXXX
XXXXXX
XST0    XST1    XST2    XST3
XXXXXX  XXXXXX  XXXXXX  XXXXXX
```

* CAUTION *

INTERPRET THE "RECORD READ" COUNT WITH CAUTION. IF IT IS VERY DIFFERENT FROM RECORD COUNT TRACKED BY THE DIAGNOSTIC, POSITION IS NOT NECESSARELY LOST. ERRORS IN READING THAT RECORD MIGHT HAVE CAUSED THE RECORD COUNT TO BE ERRONEOUSLY READ FROM TAPE.

FOR EXAMPLE, IF IN TEST 2 THE DIAGNOSTIC IS RESTARTED OR CONTINUED, THE RECORD COUNT IS RESET TO ZERO ALTHOUGH TAPE WAS NOT REWOUND. THIS IS NECESSARY BECAUSE THERE IS NO ACCURATE WAY TO DETERMINE ON WHAT RECORD COUNT OF WHAT UNIT THE DIAGNOSTIC WAS HALTED BEFORE RESTARTING OR CONTINUING. IT IS SUGGESTED THAT A "PRINT" BE REQUESTED WHEN HALTING THE DIAGNOSTIC TO GET A PRINT OF THE RECORD COUNT WHEN HALTED.

EXAMPLE OF AN ERROR REPORT:

```
CZTKIA SFT ERR 00009  TST 002  SUB 000  PC: 010606
RECOVERABLE ERROR
WRT CMD FAILED - UNIT 2  PASS:     2  RECORD:   254
PREVIOUS CMD WAS  WRT
CMDPKT  TSBA    RFC     TSSR   TCC
100005  002324  000000  100210  4
051766
000000
000371
XST0    XST1    XST2    XST3
000350  000002  100004  000000
```

3.1.1 ERROR 1 - COMMAND PACKET ADDRESS NOT ON A MODULO 4 BOUNDRY: -

IF THIS ERROR IS REPORTED, THE PROGRAM DID NOT LOAD PROPERLY. THIS IS A SYSTEM FATAL ERROR AND THE PROGRAM MUST BE RELOADED TO CORRECT IT.

3.1.2 ERROR 2 - TK25 NOT READY: -

BEFORE ANY COMMAND IS ISSUED TO THE TK25, THE SUBSYSTEM READY BIT IN THE TSSR4 IS CHECKED. IF THE SSR IS NOT SET, THE PROGRAM REPORTS THE NOT READY ERROR. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST SEQUENCE UNLESS THE IDU OPTION IS USED.

3.1.3 ERROR 3 - NO RESPONSE ERROR: -

ONCE THE TSDB IS LOADED, THE TK25 HAS ONE MILLISECOND TO RESPOND OR THE PROGRAM REPORTS A "NO RESPONSE" ERROR. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST SEQUENCE UNLESS THE IDU OPTION IS USED.

3.1.4 ERROR 4 - NO INTERRUPT ERROR: -

COMMAND WAS ISSUED AND NO INTERRUPT WAS RECEIVED. THE PROGRAM REPORTS THAT NO INTERRUPT OCCURRED. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.5 SPECIAL CONDITION ERRORS: -

IF, DURING EXECUTION, AN INCIDENT OCCURS FORCING THE TSSR SPECIAL CONDITION BIT TO SET, THE PROGRAM WILL SELECT ONE OF 8 ERROR HANDLING ROUTINES, DEPENDING ON THE TERMINATION CLASS CODE.

THE TERMINATION CLASS CODES IN THE TSSR ARE PROCESSED AS FOLLOWS WHEN SPECIAL CONDITION IS SET:

3.1.5.1 ERROR 5 - TERMINATION CLASS CODE 0, UNDEFINED SPECIAL CONDITION -

THE ERROR IS REPORTED, A HARD ERROR IS LOGGED AND THE PROGRAM PROCEEDS NORMALLY.

3.1.5.2 ERROR 6 - TERMINATION CLASS CODE 1, ATTENTION CONDITION -

THIS TCC INDICATES THAT THE DRIVE HAS UNDERGONE A STATUS CHANGE SUCH AS GOING OFFLINE OR COMING ONLINE. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.5.3 ERROR 7 - TERMINATION CLASS CODE 2, TAPE STATUS ALERT -

A STATUS CONDITION HAS BEEN ENCOUNTERED THAT MAY HAVE SIGNIFICANCE TO THE PROGRAM. BITS OF INTEREST INCLUDE TMK, RLS, LET, RLL, EOT. ACTION TAKEN DEPENDS ON THE TEST BEING EXECUTED. IF THE CONDITION IS UNEXPECTED, THE ERROR IS REPORTED AND A HARD ERROR IS LOGGED.

3.1.5.4 ERROR 8 - TERMINATION CLASS CODE 3, FUNCTION REJECT -

THE SPECIFIED FUNCTION WAS NOT INITIATED. BITS OF INTEREST INCLUDE RMR, OFL, VCK, BOT, ILC, WLE, ILA, AND NBA. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218

3.1.5.5 ERROR 9 - TERMINATION CLASS CODE 4, RECOVERABLE ERROR -

TAPE POSITION IS ONE RECORD BEYOND WHAT ITS POSITION WAS WHEN THE FUNCTION WAS INITIATED. RECOVERY PROCEDURE IS TO LOG THE ERROR AND ISSUE THE APPROPRIATE RETRY COMMAND. IF THE RETRY LIMIT IS REACHED BEFORE THE ERROR IS RECOVERED, A "RETRY LIMIT EXCEEDED" IS REPORTED AS DESCRIBED IN ERROR 14 BELOW.

3.1.5.6 ERROR 10 - TERMINATION CLASS CODE 5, RECOVERABLE ERROR -

TAPE POSITION HAS NOT CHANGED. RECOVERY PROCEDURE IS TO LOG THE ERROR AND RE-ISSUE THE ORIGINAL COMMAND. IF THE RETRY LIMIT IS REACHED BEFORE THE ERROR IS RECOVERED, A "RETRY LIMIT EXCEEDED" IS REPORTED AS DESCRIBED IN ERROR 14 BELOW.

3.1.5.7 ERROR 11 - TERMINATION CLASS CODE 6, UNRECOVERABLE ERROR -

TAPE POSITION HAS BEEN LOST. THE ONLY VALID RECOVERY PROCEDURE IS TO REWIND AND START OVER AT BOT UNLESS THE TAPE HAS LABELS OR SEQUENCE NUMBERS. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.5.8 ERROR 12 - TERMINATION CLASS CODE 7, FATAL SUBSYSTEM ERROR -

THE SUBSYSTEM IS INCAPABLE OF PROPERLY PERFORMING COMMANDS OR AT LEAST ITS INTEGRITY IS SERIOUSLY QUESTIONABLE. REFER TO THE FATAL CLASS CODE FIELD IN THE TSSR REGISTER FOR ADDITIONAL INFORMATION ON THE TYPE OF FATAL ERROR. THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.6 ERROR 13 - RFC NON-ZERO ERROR: -

IF, AFTER EXECUTION, THE RESIDUAL FRAME COUNT IS NON-ZERO, THE ERROR IS REPORTED AND A HARD ERROR IS LOGGED. THE PROGRAM THEN PROCEEDS NORMALLY. THE REPORTING AND LOGGING OF THESE ERRORS IS OPTIONAL.

3.1.7 ERROR 14 - RETRY LIMIT EXCEEDED: -

ON A WRITE COMMAND THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

ON A READ COMMAND THIS ERROR IS LOGGED AS A HARD ERROR AND THE PROGRAM PROCEEDS NORMALLY.

3.1.8 ERROR 15 - TOO MANY INTERRUPTS: -

IF MORE THAN ONE INTERRUPT OCCURS PER COMMAND, THIS ERROR IS REPORTED. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.9 ERROR 16 - CAPSTAN RUNAWAY: -

CAPSTAN DID NOT STOP WITHIN ACCEPTABLE WINDOW AFTER LAST COMMAND. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.10 ERROR 17 - DATA COMPARE ERROR: -

IF A DATA VALIDATION ERROR OCCURS DURING A WRITE/VERIFY COMMAND, THE PROGRAM PRINTS WHAT THE DATA SHOULD HAVE BEEN AND WHAT THE DATA WAS, AND PRINTS THE BYTE AND RECORD NUMBER THE ERROR OCCURRED ON. ONLY THE FIRST 10 BYTES IN ERROR PER RECORD ARE PRINTED. THE TOTAL NUMBER OF BYTES IN ERROR PER RECORD IS ALSO PRINTED. A HARD ERROR IS LOGGED AND THE PROGRAM PROCEEDS NORMALLY.

3.2 ERROR HALTS

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

```
1267
1268
1269
1270                          4.0  PERFORMANCE REPORT
1271
1272
1273                               UNIT X   PASS:XXXXX  RECORD:XXXXX
1274                               BYTES WRITTEN  XXX,XXX,XXX,XXX
1275                               BYTES READ REV XXX,XXX,XXX,XXX
1276                               BYTES READ FWD XXX,XXX,XXX,XXX
1277                                                 WRT       RDR     RDF
1278                               RECOVERABLE ERRORS    XXXXX     XXXXX   XXXXX
1279                               UNRECOVERABLE ERRORS  XXXXX     XXXXX   XXXXX
1280
1281                               SPEC COND   HARD   FATAL   COMPARE
1282                                  XXXXX   XXXXX   XXXXX   XXXXX
1283
1284
1285
1286
1287                          5.0  TEST SUMMARIES
1288
1289                          5.1  TEST 1 - BASIC FUNCTIONS.
1290
1291                          EXECUTES AND VERIFIES CORRECT COMPLETION OF ALL TK25 FUNCTIONS.
1292
1293                                SUBTEST 1  - SET CHAR, DRIVE INIT, GET STATUS.
1294                                       * SET CHARACTERISTIC 200. (ENABLES SKIP TAPE MARKS STOP)
1295                                       * DRIVE INITIATE.
1296                                       * SET CHARACTERISTIC  20. (ENABLES MESSAGE BUFF RELEASE INTERRUPTS)
1297                                       * GET STATUS
1298                                       * SET CHARACTERISTIC  40. (ENABLES ATTENTION INTERRUPTS)
1299
1300                                SUBTEST 2  - REWIND.
1301                                       * REWIND.
1302                                       * REWIND AT BOT.
1303
1304                                SUBTEST 3  - WRITE/VERIFY.
1305                                       * WRITE/VERIFY PATTERN 1.
1306                                       * WRITE/VERIFY PATTERN 2.
1307                                       * WRITE/VERIFY PATTERN 3.
1308                                       * WRITE/VERIFY PATTERN 4.
1309                                       * WRITE/VERIFY PATTERN 5.
1310                                       * WRITE/VERIFY PATTERN 6.
1311                                       * WRITE/VERIFY PATTERN 0.
1312
1313                                SUBTEST 4  - WRITE TAPE MARK, ERASE.
1314                                       * WRITE TAPE MARK.
1315                                       * WRITE 10 RECORDS
1316                                       * ERASE 10 TIMES
1317                                       * WRITE TAPE MARK.
1318                                       * WRITE TAPE MARK RETRY.
1319
1320                                SUBTEST 5  - SPACE FILES.
1321                                       * SPACE 2 FILES REVERSE.
```

```
1323
1324
1325
1326                                    + SPACE 2 FILES FORWARD.
1327                                    + SPACE 2 FILES REVERSE.
1328                                    + SPACE 2 FILES FORWARD.
1329
1330                                SUBTEST 6  - SPACE RECORDS.
1331                                    + REWIND.
1332                                    + SPACE 7 RECORDS FORWARD.
1333                                    + SPACE 7 RECORDS REVERSE.
1334                                    + SPACE 7 RECORDS FORWARD.
1335                                    + SPACE 7 RECORDS REVERSE.
1336
1337                                SUBTEST 7  - WRITE RETRY.
1338                                    + REWIND.
1339                                    + WRITE DATA.
1340                                    + WRITE RETRY.
1341
1342                                SUBTEST 8  - READ REV RETRY.
1343                                    + READ REVERSE.
1344                                    + READ NEXT REVERSE.
1345                                    + READ NEXT FORWARD.
1346
1347                                SUBTEST 9  - READ FWD RETRY.
1348                                    + READ FORWARD.
1349                                    + READ PREVIOUS FORWARD.
1350                                    + READ PREVIOUS REVERSE.
1351
1352                                SUBTEST 10 - CLEAN.
1353                                    + CLEAN.
1354                                    + REWIND.
1355
1356                                SUBTEST 11  - WRITE/VERIFY SWAPPED DATA BYTES.
1357                                    + WRITE/VERIFY EVEN LENGTH (RECORD 1).
1358                                    + WRITE/VERIFY ODD LENGTH (RECORD 2).
1359                                    + SET DATA BYTE SWAP.
1360                                    + WRITE/VERIFY EVEN LENGTH (RECORD 3).
1361                                    + WRITE/VERIFY ODD LENGTH (RECORD 4).
1362                                    + CLEAR DATA BYTE SWAP.
1363
1364                                SUBTEST 12  - READ SWAPPED DATA BYTES.
1365                                    + READ REV RECORD 4.
1366                                    + READ REV RECORD 3.
1367                                    + SET DATA BYTE SWAP.
1368                                    + READ REV RECORD 2.
1369                                    + READ REV RECORD 1.
1370                                    + READ FWD RECORD 1.
1371                                    + READ FWD RECORD 2.
1372                                    + CLEAR DATA BYTE SWAP.
1373                                    + READ FWD RECORD 3.
1374                                    + READ FWD RECORD 4.
1375
```

5.2 TEST 2 - DATA RELIABILITY.

1. THE TAPE IS INITIATED WITH THE FOLLOWING COMMANDS:

   SET CHARACTERISTIC 40
   REWIND
   WRITE 31 RECORDS OF RANDOM LENGTH AND DATA

2. WRITE, REPOSITION RECORD REVERSE, AND READ COMMANDS ARE SELECTED AT RANDOM AND ARE EXECUTED A RANDOM NUMBER OF TIMES WITH RANDOM LENGTHS AND RANDOM PATTERN UNTIL END OF TAPE IS REACHED.

3. AT THE END OF EACH PASS, A REWIND COMMAND IS ISSUED AND A PERFORMANCE REPORT IS PRINTED.

NOTE

IF A RESTART COMMAND IS USED TO INITIATE TEST 1, THE INITIAL REWIND COMMAND IS NOT ISSUED.

5.3 TEST 3 - WRITE AND READ STREAMING TEST.

****
*** NOTE: THE TAPE LENGTH MUST BE 600 FEET FOR THIS TEST ***
****

1. REWINDS ALL UNITS, THEN ON EACH UNIT:

****

2. WRITE PATTERN 5 FOR 11719 - 4096 BYTE RECORDS.
3. SPACE REVERSE FOR 11719 RECORDS.
4. READ FORWARD FOR 11719 RECORDS.

****

NOTE

WRITE AND READ ITERATIONS ARE INTERRUPTED SO THAT THE TAPE DOES NOT CONTINUOUSLY STREAM. THREE RECORDS ARE ALLOWED TO STREAM BEFORE THE UNIT IS INTERRUPTED FOR BOTH WRITE AND READ OPERATIONS. THE INTERRUPTION SCHEME IS SET UP TO PERMIT STREAMING OPERATIONS 75% OF THE TIME THIS TEST IS IN EXECUTION.

THIS TEST EXECUTES IN A SINGLE SERVER MANNER - ONLY ONE UNIT AT A TIME IS EVER IN OPERATION. FOR A FOUR (4) UNIT CONFIGURATION, THIS MEANS THAT THE DUTY CYCLE OF THE SYSTEM WILL BE 25%, BUT THE STREAMING OPERTATION OF THE SPECIFIC UNIT UNDER TEST WILL BE 75%. FOR A THREE (3) UNIT CONFIGURATION THE SYSTEM DUTY CYCLE WILL BE 33.3%, AND A TWO UNIT CONFIGURATION WILL BE 50%. THE STREAMING DUTY CYCLE FOR ANY PARTICULAR UNIT UNDER TEST WILL ALWAYS BE 75% REGARDLESS OF THE SYSTEM DUTY CYCLE.

5.4 TEST 4 - WRITE COMPATABILITY/WRITE UTILITY.

REWINDS AND WRITES RECORDS OF RANDOM LENGTHS AND RANDOM DATA FROM BOT TO EOT.

5.5 TEST 5 - READ COMPATABILITY/READ UTILITY.

REWINDS AND READS ENTIRE TAPE IN THE FORWARD DIRECTION, REWIND.

5.6 TEST 6 - EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.

THE SEQUENCE OF COMMANDS ENTERED BY THE OPERATOR IS EXECUTED. IF NO COMMANDS WERE ENTERED, A DEFAULT SEQUENCE OF REWIND/WRITE/REWIND/READ FWD/REWIND OF THE ENTIRE TAPE IS EXECUTED WITH RANDOM PATTERN AND RECORD LENGTH OF 4096 BYTES.

6.0 DEVICE INFORMATION TABLES

6.1 GENERAL

THE TK25 SUBSYSTEM IS A 1/4 INCH CARTRIDGE TAPE DRIVE WITH EITHER A UNIBUS OR Q-BUS CONTROLLER. THE CONTROLLER MODULES USE THE TS11 PROTOCOL AS DEFINED IN THE TK25 SUBSYSTEM PROGRAMMERS GUIDE.

COMMANDS ARE NOT WRITTEN TO THE DRIVE; RATHER, COMMAND POINTERS ARE WRITTEN WHICH POINT TO COMMAND PACKETS SOMEWHERE IN CPU MEMORY. THE COMMAND POINTER IS USED BY THE TK25 SUBSYSTEM TO FETCH THE WORD(S) WITHIN THE COMMAND PACKET. THE WORDS WITHIN THE COMMAND PACKET ARE:

1. COMMAND WORD
2. LOW ORDER BUFFER ADDRESS
3. HIGH ORDER BUFFER ADDRESS
4. BYTE COUNT

THE TSSR CONTAINS ALL THE INFORMATION WHICH WILL BE NECESSARY TO DETERMINE WHETHER:

1. THE DRIVE IS READY TO ACCEPT ANOTHER COMMAND,
2. THE PREVIOUS COMMAND WAS EXECUTED WITHOUT ERROR.

IF EITHER OF THE ABOVE CONDITIONS IS UNTRUE AT "JOB DONE" OR "COMMAND INITIATION" TIME, IT MAY BE NECESSARY TO GET THE EXTENDED STATUS REGISTERS TO DETERMINE WHAT ACTION IS TO BE TAKEN AND/OR LOG THE ERROR INFORMATION.

EXTENDED STATUS REGISTERS ARE NOT READ DIRECTLY FROM DRIVE REGISTERS; RATHER, A "GET STATUS" COMMAND IS ISSUED WHICH WILL CAUSE THE TK25 TO TRANSFER EXTENDED STATUS INFORMATION TO THE MEMORY AREA POINTED TO BY THE BUFFER ADDRESS OF THE "GET STATUS" COMMAND. THERE ARE FOUR EXTENDED STATUS REGISTERS.

THE TSDB MUST BE WRITTEN WITH A DATO INSTRUCTION TO PROPERLY WRITE THE COMMAND POINTER. A DATOB WILL CAUSE A MAINTENANCE FUNCTION. A DATO TO THE TSSR WILL CAUSE SUBSYSTEM INIT.

COMMAND PACKETS MUST RESIDE ON DIVIDE BY FOUR MEMORY BOUNDARIES (AS OPPOSED TO DIVIDE BY 2 OR WORD BOUNDARIES).

6.2 BUS INTERFACE SPECIFICATIONS

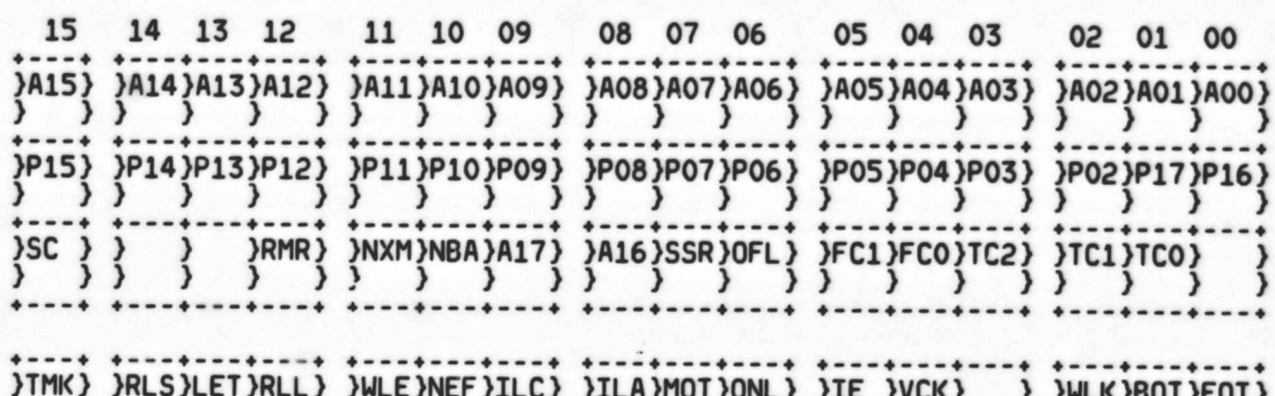
THE STANDARD ADDRESSES AND VECTORS ARE ASSIGNED AS FOLLOWS:

TK25	REGISTER	Q-BUS ADDRESS (I/O PAGE) BBS7		UNIBUS ADDRESS	VECTOR
0	TSBA/TSDB	1	2 520	772 520	
	TSSR	1	2 522	772 522	224
1	TSBA/TSDB	1	2 524	772 524	
	TSSR	1	2 526	772 526	*
2	TSBA/TSDB	1	2 530	772 530	
	TSSR	1	2 532	772 532	*
3	TSBA/TSDB	1	2 534	772 534	
	TSSR	1	2 536	772 536	*

THE "*" INDICATES THAT THE VECTOR IS A FLOATING VECTOR WITH RANK OF 37. FLOATING VECTORS ARE ASSIGNED STARTING AT 300 WITH THE LOWER RANK NUMBERS BEING ASSIGNED FIRST. NOTE THAT THE FLOATING VECTORS MAY CHANGE FROM SYSTEM TO SYSTEM.

6.3 BIT DEFINITIONS FOR TK25 REGISTERS

6.3.1 TK25 REGISTER SUMMARY -

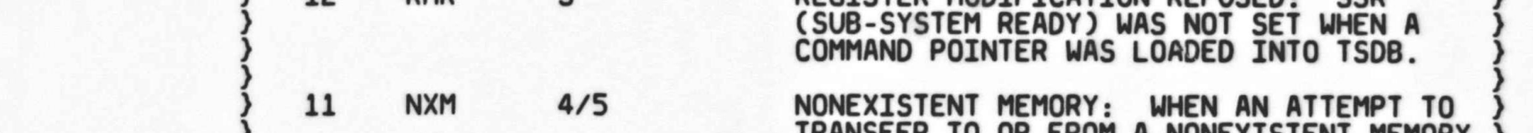


TERMINATION CLASS CODES (TSSR TC0-TC2):

0	=	NORMAL TERMINATION
1	=	ATTENTION CONDITION
2	=	TAPE STATUS ALERT
3	=	FUNCTION REJECT
4	=	RECOVERABLE ERROR - TAPE POSITION = ONE RECORD DOWN TAPE FROM START OF FUNCTION
5	=	RECOVERABLE ERROR - TAPE NOT MOVED
6	=	UNRECOVERABLE ERROR - TAPE POSITION LOST
7	=	FATAL CONTROLLER ERROR

6.3.2 TK25 STATUS REGISTER (TSSR) -

6.3.2.1 TSSR READ ONLY -



BIT	NAME	TERMINATION CLASS	DEFINITION
15	SC	S	SPECIAL CONDITION: USED TO INDICATE THAT EITHER AN ERROR OR AN EXCEPTION OCCURRED WHILE EXECUTING A COMMAND.
14	-	-	NOT USED
13	-	-	NOT USED
12	RMR	S	REGISTER MODIFICATION REFUSED: SSR (SUB-SYSTEM READY) WAS NOT SET WHEN A COMMAND POINTER WAS LOADED INTO TSDB.
11	NXM	4/5	NONEXISTENT MEMORY: WHEN AN ATTEMPT TO TRANSFER TO OR FROM A NONEXISTENT MEMORY LOCATION, THIS BIT WILL SET.
10	NBA	S	NEED BUFFER ADDRESS: INDICATES THAT THE MESSAGE BUFFER ADDRESS IS REQUIRED.
09	A17	S	BUS ADDRESS BIT 17: DISPLAYS VALUE OF BIT 17 OF THE TSBA REGISTER.
08	A16	S	BUS ADDRESS BIT 16: DISPLAYS VALUE OF BIT 16 OF THE TSBA REGISTER.

BIT	NAME	TERMINATION CLASS	DEFINITION
07	SSR	S	SUBSYSTEM READY: THE SUBSYSTEM IS NOT BUSY AND IS READY TO ACCEPT A NEW COMMAND POINTER WHENEVER THIS BIT IS SET.
06	OFL	S	OFF LINE: INDICATES THE TAPE TRANSPORT IS OFF LINE AND UNAVAILABLE FOR ANY MOTION COMMANDS.
05	FC1	-	FATAL TERMINATION CLASS 1 - NOT USED
04	FC0	-	FATAL TERMINATION CLASS 0 - NOT USED
03	TC2	S	TERMINATION CLASS BIT 2: THIS BIT, ALONG WITH TC1 AND TC0, ACTS AS AN OFFSET VALUE WHENEVER AN ERROR OR EXCEPTION CONDITION OCCURS ON A COMMAND. EACH OF THE EIGHT POSSIBLE VALUES OF THIS FIELD REPRESENTS A PARTICULAR CLASS OF ERRORS WHICH HAVE A SIMILAR SIGNIFICANCE AND, AS APPLICABLE, SIMILAR RECOVERY PROCEDURES. THE CODE PROVIDED IN THIS FIELD IS EXPECTED TO BE USED AS AN OFFSET INTO A DISPATCH TABLE FOR HANDLING OF THE CONDITION.
02	TC1	S	TERMINATION CLASS BIT 1: SEE TC2
01	TC0	S	TERMINATION CLASS BIT 0: SEE TC2
00	-		NOT USED

6.3.2.2 TSSR WRITE ONLY -

WRITE COMMANDS TO THE TSSR DO NOT WRITE, BUT INVOKE CERTAIN SPECIALIZED FUNCTIONS.

1. A WRITE WORD (DATO) TO THE TSSR WILL INITIALIZE THE TK25 SUBSYSTEM AND REWIND THE TAPE ON THE CARTRIDGE.

2. A WRITE BYTE (DATOB) TO THE LOW BYTE OF THE TSSR WILL CAUSE THE CONTROLLER TO EXECUTE ITS RESIDENT SELF TESTS. IF THE CONTROLLER PASSES THE SELF TESTS, IT WILL THEN INITIALIZE ITSELF AND THE DRIVE AS ABOVE. THE TSDB/BA AND THE TSSR WILL NOT RESPOND TO BUS TRANSACTIONS FOR THE DURATION OF THE SELF TEST (APPROXIMATELY 100 MICROSECONDS). ANY ATTEMPT BY THE HOST TO READ OR WRITE THESE REGISTERS DURING SELF TEST WILL RESULT IN A NON-EXISTENT DEVICE REGISTER TIMEOUT.

3. IF AN OPERATION IS NOT IN PROGRESS (SSR SET IN THE TSSR), A WRITE BYTE TO THE HIGH BYTE OF THE TSSR WITH A "1" IN THE HIGH ORDER DATA BIT POSITION (BIT 7) WILL CAUSE THE SUBSYSTEM TO BOOT THE CPU BY MEANS OF THE FOLLOWING SEQUENCE OF EVENTS (Q-BUS CONTROLLER ONLY):

   - REWIND THE TAPE
   - SPACE FORWARD ONE RECORD
   - READ THE FIRST 256 BITS OF THE SECOND RECORD INTO CPU MEMORY STARTING AT ADDRESS 0.

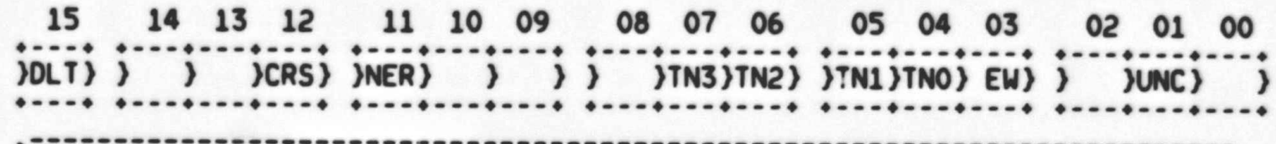
6.3.3 EXTENDED STATUS REGISTER 0 (XSTAT0) -

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
XST0	TMK	RLS	LET	RLL	WLE	NEF	ILC	ILA	MOT	ONL	IE	VCK		WLK	BOT	EOT

BIT	NAME	TERMINATION CLASS	DEFINITION
15	TMK	S/2	TAPE MARK DETECTED: SET WHENEVER A TAPE MARK IS DETECTED ON TAPE.
14	RLS	2	RECORD LENGTH SHORT: USED TO INDICATE ONE OF THREE CONDITIONS: 1. THE RECORD READ WAS SHORTER THAN THE BYTE COUNT 2. A SPACE RECORD OPERATION TERMINATED BEFORE THE POSITION COUNT WAS COMPLETED (THIS IS NORMAL IF A TAPE MARK IS DETECTED OR BOT IS ENCOUNTERED IN A SPACE REVERSE). 3. A SKIP TAPE MARKS COMMAND TERMINATED BEFORE THE POSITION COUNT WAS COMPLETED. THIS IS NORMAL IF A DOUBLE TAPE MARK (SEE LET) IS DETECTED OR BOT IS ENCOUNTERED IN A REVERSE OPERATION.
13	LET	2	LOGICAL END OF TAPE: SETS IF A TAPE MARK IS DETECTED WHEN MOVING FROM BOT OR IF DOUBLE TAPE MARKS ARE DETECTED. NOTE: THIS BIT WILL ONLY SET IF THE COMMAND IS SKIP TAPE MARKS AND THE MODE OF TERMINATION WAS ENABLED BY THE SET CHARACTERISTICS COMMAND.
12	RLL	2	RECORD LENGTH LONG: THE RECORD READ WAS LONGER THAN THE BYTE COUNT SPECIFIED.
11	WLE	3,(6)	WRITE LOCK ERROR: THE WRITE OPERATION WAS ISSUED TO A WRITE PROTECTED TAPE. NOTE: THE TK25 SETS THE TERMINATION CODE 3 ONLY. TC 6 APPLIES ONLY TO DRIVES EQUIPPED WITH A WRITE LOCK SWITCH
10	NEF	3	NON-EXECUTABLE FUNCTION: CAUSED TO SET BY ONE OF THE FOLLOWING: . REVERSE COMMAND WHEN ALREADY AT BOT

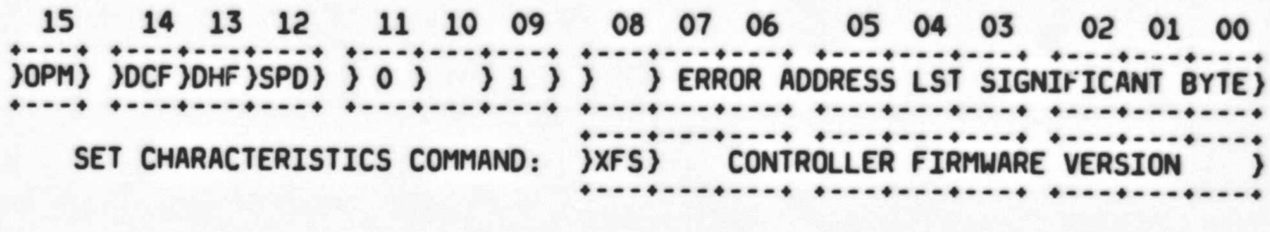
			. ANY MOTION COMMAND WITHOUT A CLEAR VOLUME CHECK WHILE THE TRANSPORT WAS OFF-LINE. . A WRITE COMMAND TO A WRITE PROTECTED TAPE.
09	ILC	3	ILLEGAL COMMAND: ANY COMMAND THAT CONTAINS CODES IN EITHER THE COMMAND FIELD OR MODE FIELD THAT ARE NOT SUPPORTED BY THE TAPE TRANSPORT.
08	ILA	3	ILLEGAL ADDRESS: WHEN THIS BIT IS SET, AN ILLEGAL ADDRESS IS ISSUED.
07	MOT	S	CAPSTAN MOVED: INDICATES TAPE WAS MOVED DURING AN OPERATION.
06	ONL	5/1/3	ON-LINE: INDICATES THAT THE SELECTED TRANSPORT IS ON-LINE AND READY. TERMINATION CLASS 1 IS SET FOR ATTN INTERRUPTS AND TC 3 FOR NON-EXECUTABLE FUNCTIONS REJECTED WHILE OFF LINE.
05	IE	S	INTERRUPT ENABLE:
04	VCK	S/3	VOLUME CHECK: ALWAYS SET AFTER INITIALIZATION OR WHEN THE ON-LINE STATUS OF THE TRANSPORT CHANGES. (EITHER ON TO OFF OR OFF TO ON)
03	PED	S	PHASE ENCODED DRIVE: TK25 IS NOT P.E., SO THIS BIT IS ALWAYS ZERO.
02	WLK	S/3	WRITE LOCKED: THE TAPE IS WRITE PROTECTED.
01	BOT	S/2/3	BEGINNING OF TAPE: THE TAPE IS AT LOAD POINT, TC2 IS SET IF TAPE IS REVERSED INTO BOT, AND TC3 IF A REVERSE COMMAND IS ISSUED WITH THE TAPE ALREADY A BOT.
00	EOT	S/2	END OF TAPE: SETS AS THE HEAD PASSES TO TRACK 10 IN THE FORWARD DIRECTION. IT IS NOT RESET UNTIL THE HEAD PASSES BACK TO TRACK 9 IN THE REVERSE DIRECTION. (STATUS ON READ; TC2 ON WRITE. SUBSYSTEM INIT ALSO RESETS THIS BIT.

6.3.4 EXTENDED STATUS REGISTER 1 (XSTAT1) -



BIT	NAME	TERMINATION CLASS	DEFINITION
15	DLT	4	DATA LATE: THIS BIT IS SET WHEN THE 16 BYTE FIFO BUFFER IS FULL ON A READ, OR EMPTY ON A WRITE, AND THE TAPE TRANSPORT REQUIRES A DATA TRANSFER.
14	-	-	NOT USED.
13	-	-	NOT USED.
12	CRS	4	CREASE DETECTED. DATA DROPPED OUT FOR UP TO 1.8 MS (APPROXIMATELY 0.2 INCH).
11	NER	4	NOISE DETECTED DURING ERASE.
10	-	-	NOT USED.
09	-	-	NOT USED.
08	-	-	NOT USED.
07	TN3	S	TAPE TRACK NUMBER HIGH ORDER BIT.
06	TN2	S	TAPE TRACK NUMBER BIT 2.
05	TN1	S	TAPE TRACK NUMBER BIT 1.
04	TN0	S	TAPE TRACK NUMBER LOW ORDER BIT.
03	EW	S/4	EARLY WARNING HOLE AT END OF TRACK SEEN.
02	-	-	NOT USED.
01	UNC	4	UNCORRECTABLE DATA: IN THE TK25 ALL TAPE ERRORS ARE UNCORRECTABLE, SINCE THERE IS NO INTERNAL ERROR CORRECTION.
00	-	-	NOT USED.

6.3.5 EXTENDED STATUS REGISTER 2 (XSTAT2) -



BIT	NAME	TERMINATION CLASS	DEFINITION
15	OPM	S	OPERATION IN PROGRESS: TAPE MOVED
14	DCF	7	DRIVE COMMUNICATION FAULT. FAILURE OF READ SENSE COMMAND TO TCS INITIATED AFTER DETECTING INT OR DER RESULTING FROM TAPE MOTION COMMAND TO TCS; OR HEALTH CHECK FAULT RECEIVED IN TCS SENSE BYTES.
13	DHF	7	DRIVE HARDWARE FAULT. NO LAMP CURRENT STATUS OR HEAD POSITIONING FAULT RETURNED IN TCS SENSE BYTES.
12	SPD	7	CAPSTAN SPEED ERROR, FAST OR SLOW
11	0	-	TK25 IDENTIFIER. ALWAYS ZERO.
10	-	-	NOT USED.
09	1	-	TK25 IDENTIFIER. ALWAYS ONE.
08	-	-	NOT USED.
07-00	EAD 07-00	S	ERROR ADDRESS LEAST SIGNIFICANT BYTE. (PROGRAM COUNTER IN THE TCD MICROCODE.)

NOTE: XSTAT 2 BITS 07 THRU 00 HAVE A DIFFERENT MEANING DURING THE SET CHARACTERISTICS COMMAND: XSTAT 2 BITS 06 THRU 00 RETURN THE MAJOR REVISION LEVEL OF THE CONTROLLER MICROCODE (IN BINARY). IF BIT 07 IS A 1, THE CONTROLLER ASSUMES A 22 BIT Q-BUS, AND IF 0, AN 18 BIT Q-BUS OR A UNIBUS. IN THE LATTER CASE, COMMAND PACKETS CONTAINING ADDRESSES GREATER THAN 18 BITS WILL GENERATE ERRORS.

6.3.6 EXTENDED STATUS REGISTER 3 (XSTAT3) -

15	14 13 12	11 10 09	08 07 06	05 04 03	02 01 00
ERROR ADDRESS MOST SIGNIFICANT BYTE			OPI	REV TCH STP	RIB

BIT	NAME	TERMINATION CLASS	DEFINITION
15-08	EAD 15-08	S	ERROR ADDRESS MOST SIGNIFICANT BYTE. IF XSTAT 1 BIT 14 IS 1, THE ADDRESS IS THE PROGRAM COUNTER IN THE SIA, AND IF 0 IN THE TCS.
07	-	-	NOT USED.
06	OPI	6	OPERATION INCOMPLETE: SETS IF 16 FEET OF TAPE WITHOUT DATA PASSES THE READ HEAD ON READ, SPACE, OR SKIP OPERATIONS OR 4 FEET OF TAPE IN WRITE OPERATIONS.
05	REV	S	REVERSE: INDICATES REVERSE IS THE DIRECTION OF CURRENT TAPE OPERATION.
04	TCH	7	NO TACHS: INDICATES THAT CAPSTAN MOTOR DID NOT START OR THAT TACHS ARE NOT BEING GENERATED OR DETECTED.
03	STP	S/6	STRIPE: SERVO STRIPE IS FAULTY OR MISSING.
02	-	-	NOT USED.
01	-	-	NOT USED.
00	RIB	2	REVERSE INTO BOT: SETS WHEN REVERSE OPERATIONS ENCOUNTER THE BOT EARLY WARNING HOLE AFTER TAPE IS IN MOTION.

ε

```
                                .TITLE PROGRAM HEADER AND TABLES
                                .SBTTL  PROGRAM HEADER

000000                                  .ENABL  ABS,AMA
002000                          .       =       2000
002000                                  BGNMOD

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

002000                                  POINTER BGNRPT,BGNSW,BGNSFT,BGNAU,BGNDU,BGNSETUP

002000                                  HEADER  CZTKI,B,0,5000,1,INTPRI
002000                          L$NAME::                ;DIAGNOSTIC NAME
002000  103                                                             .ASCII /C/
002001  132                                                             .ASCII /Z/
002002  124                                                             .ASCII /T/
002003  113                                                             .ASCII /K/
002004  111                                                             .ASCII /I/
002005  000                                                             .BYTE   0
002006  000                                                             .BYTE   0
002007  000                                                             .BYTE   0
002010                          L$REV::                 ;REVISION LEVEL
002010  102                                                             .ASCII  /B/
002011                          L$DEPO::                ;0
002011  060                                                             .ASCII  /0/
002012                          L$UNIT::                ;NUMBER OF UNITS
002012  000001                                                          .WORD   T$PTHV
002014                          L$TIML::                ;LONGEST TEST TIME
002014  005000                                                          .WORD   5000
002016                          L$HPCP::                ;POINTER TO H.W. QUES.
002016  030474                                                          .WORD   L$HARD
002020                          L$SPCP::                ;POINTER TO S.W. QUES.
002020  030546                                                          .WORD   L$SOFT
002022                          L$HPTP::                ;PTR. TO DEF. H.W. PTABLE
002022  002176                                                          .WORD   L$HW
002024                          L$SPTP::                ;PTR. TO S.W. PTABLE
002024  002204                                                          .WORD   L$SW
002026                          L$LADP::                ;DIAG. END ADDRESS
002026  032234                                                          .WORD   L$LAST
002030                          L$STA::                 ;RESERVED FOR APT STATS
002030  000000                                                          .WORD   0
002032                          L$CO::
002032  000000                                                          .WORD   0
002034                          L$DTYP::                ;DIAGNOSTIC TYPE
002034  000001                                                          .WORD   1
002036                          L$APT::                 ;APT EXPANSION
002036  000000                                                          .WORD   0
002040                          L$DTP::                 ;PTR. TO DISPATCH TABLE
002040  002124                                                          .WORD   L$DISPATCH
002042                          L$PRIO::                ;DIAGNOSTIC RUN PRIORITY
002042  000340                                                          .WORD   INTPRI
002044                          L$ENVI::                ;FLAGS DESCRIBE HOW IT WAS SETUP
002044  000000                                                          .WORD   0
```

PROGRAM HEADER

SEQ 0044

```
002046
002046  000000                  L$EXP1::        ;EXPANSION WORD
002050                                                          .WORD   0
002050  003                     L$MREV::        ;SVC REV AND EDIT #
002051  003                                                     .BYTE   C$REVISION
002052                                                          .BYTE   C$EDIT
002052  000000                  L$EF::          ;DIAG. EVENT FLAGS
002054  000000                                                  .WORD   0
002056                          L$SPC::                         .WORD   0
002056  000000
002060                          L$DEVP::        ; POINTER TO DEVICE TYPE LIST
002060  002166                                                  .WORD   L$DVTYP
002062                          L$REPP::        ;PTR. TO REPORT CODE
002062  020440                                                  .WORD   L$RPT
002064                          L$EXP4::
002064  000000                                                  .WORD   0
002066                          L$EXP5::
002066  000000                                                  .WORD   0
002070                          L$AUT::         ;PTR. TO ADD UNIT CODE
002070  024424                                                  .WORD   L$AU
002072                          L$DUT::         ;PTR. TO DROP UNIT CODE
002072  024360                                                  .WORD   L$DU
002074                          L$LUN::         ;LUN FOR EXERCISERS TO FILL
002074  000000                                                  .WORD   0
002076                          L$DESP::        ;POINTER TO DIAG. DESCRIPTION
002076  002140                                                  .WORD   L$DESC
002100                          L$LOAD::        ;GENERATE SPECIAL AUTOLOAD EMT
002100  104035                                                  EMT     E$LOAD
002102                          L$ETP::         ;POINTER TO ERRTBL
002102  000000                                                  .WORD   0
002104                          L$ICP::         ;PTR. TO INIT CODE
002104  022252                                                  .WORD   L$INIT
002106                          L$CCP::         ;PTR. TO CLEAN-UP CODE
002106  024302                                                  .WORD   L$CLEAN
002110                          L$ACP::         ;PTR. TO AUTO CODE
002110  023660                                                  .WORD   L$AUTO
002112                          L$PRT::         ;PTR. TO PROTECT TABLE
002112  022244                                                  .WORD   L$PROT
002114                          L$TEST::        ;TEST NUMBER
002114  000000                                                  .WORD   0
002116                          L$DLY::         ;DELAY COUNT
002116  000000                                                  .WORD   0
002120                          L$HIME::        ;PTR. TO HIGH MEM
002120  000000                                                  .WORD   0
2054
```

```
                                .SBTTL  DISPATCH TABLE

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

                                        DISPATCH 6              ; SIX TESTS
002122  000006                                                          .WORD   6
                                L$DISPATCH::
002124  024520                                                          .WORD   T1
002126  026106                                                          .WORD   T2
002130  026562                                                          .WORD   T3
002132  027250                                                          .WORD   T4
002134  027414                                                          .WORD   T5
002136  027526                                                          .WORD   T6

                                .SBTTL  DESCRIPTIVE TEXT

                                ;++
                                ;2 LINES OF TEXT PRINTED TO THE OPERATOR TO IDENTIFY THE DIAGNOSTIC AND THE DEVICE UNDER TES
                                ;--

                                        DESCRIPT        <DATA RELIABILITY TEST>
                                L$DESC::
002140  104     101     124                                             .ASCIZ  /DATA RELIABILITY TE
002143  101     040     122
002146  105     114     111
002151  101     102     111
002154  114     111     124
002157  131     040     124
002162  105     123     124
002165  000

                                                                        .EVEN
                                        DEVTYP          <TK25>
                                L$DVTYP::
002166  124     113     062                                             .ASCIZ  /TK25/
002171  065     000
                                                                        .EVEN
```

```
                        .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 RUN-TIME P-TABLE.
                        ;--

002174                          BGNHW   DFPTBL
002174  000002                                                  .WORD   L10000-L$HW/2
002176                  L$HW::
002176                  DFPTBL::

002176  172522          172522          ;TSSR ADDRESS.
002200  000224          224             ;VECTOR ADDRESS.

002202                          ENDHW
002202                  L10000:
```

```
                                .SBTTL  SOFTWARE P-TABLE

                                ;++
                                ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
                                ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
                                ;--
                                        BGNSW   SFPTBL
002202  000043                                                      .WORD   L10001-L$SW/2
002204                          L$SW::
002204                          SFPTBL::
002204  001                     CLRFLG::.BYTE   1               ;CLEAR COUNTERS FLAG.
002205  000                     RRANV:: .BYTE   0               ;RESET RANDOM VARIABLES EACH PASS FLAG.
002206  000                     HAE::   .BYTE   0               ;HALT AFTER EACH COMMAND FLAG.
002207  000                     ERCVER::.BYTE   0               ;ENABLE RECOVERABLE ERROR PRINTS FLAG.
002210  000                     IREC::  .BYTE   0               ;INHIBIT ERROR RECOVERY FLAG.
002211  001                     BADTSW::.BYTE   1               ;BAD TAPE SWITCH TO REWRITE ON SAME SPOT & DETECT BAD TAPE
002212  000                     DINT::  .BYTE   0               ;DISABLE INTERRUPTS FLAG.
002213  000                     PIRE::  .BYTE   0               ;INHIBIT RESIDUAL FRAMECOUNT ERROR REPORT FLAG.
002214  000                     RAMWRT::.BYTE   0               ;ENABLE OPTIONAL RAM DUMP
002215  000                             .BYTE   0               ;SPARE
002216  000                     EWPRNT::.BYTE   0               ;ENABLE EARLY WARNING END OF TRACK STATUS PRINTS
002217  000                     CHGFLG::.BYTE   0               ;CHANGE CMD SEQ TABLE FLAG.
002220                                  .EVEN
002220  000040                  CHAR::  CH.EAI                  ;CHARACTERISTICS CODE (DEFAULT = 40).
002222  000015                  CMDD::  .WORD   13.             ;COMMAND 2 (DEFAULT = REWIND).
002224  000001                          .WORD   1               ;BYTE COUNT
002226  000001                          .WORD   1               ;NUMBER OF OPERATIONS
002230  000007                          .WORD   RANP            ;RANDOM DATA
002232  000004                          .WORD   4               ;COMMAND 3  (DEFAULT = WRITE)
002234  010000                          .WORD   DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
002236  035230                          .WORD   15000.          ;NUMBER OF OPERATIONS (DEFAULT = 15000).
002240  000007                          .WORD   RANP            ;RANDOM DATA
002242  000015                          .WORD   13.             ;COMMAND 4 (DEFAULT = REWIND)
002244  000001                          .WORD   1               ;BYTE COUNT
002246  000001                          .WORD   1               ;ONE OPERATION
002250  000007                          .WORD   RANP            ;RANDOM DATA
002252  000002                          .WORD   2               ;COMMAND 5 (DEFAULT = READ FWD).
002254  010000                          .WORD   DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
002256  035230                          .WORD   15000.          ;NUMBER OF OPERATIONS (DEFAULT = 15000)
002260  000007                          .WORD   RANP            ;RANDOM DATA
002262  000033                          .WORD   27.             ;TERMINATOR
002264  000001                          .WORD   1               ;BYTE COUNT
002266  000001                          .WORD   1               ;NUMBER OF OPERATIONS
002270  000007                          .WORD   RANP            ;PATTERN
002272  000033                          .WORD   27.             ;END OF CMD SEQ TABLE CODE (DEF) OR CMD 7
002274  010000                          .WORD   DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
002276  035230                          .WORD   15000.          ;NUMBER OF OPERATIONS (DEFAULT = 15000).
002300  000007                          .WORD   RANP            ;RANDOM DATA
002302  000033                          .WORD   27.             ;END OF CMD SEQ TABLE CODE (DEF) OR CMD 8
002304  010000                          .WORD   DATCNT          ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
002306  035230                          .WORD   15000.          ;NUMBER OF OPERATIONS (DEFAULT = 15000).
002310  000007                          .WORD   RANP            ;RANDOM DATA
002312                                  ENDSW
002312                          L10001:
002312                                  ENDMOD
```

```
                                .TITLE GLOBAL AREAS
                                .SBTTL  GLOBAL EQUATES SECTION

002312                                  BGNMOD

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

002312                                  EQUALS
                                ;
                                ; BIT DIFINITIONS
                                ;
100000                          BIT15== 100000
040000                          BIT14== 40000
020000                          BIT13== 20000
010000                          BIT12== 10000
004000                          BIT11== 4000
002000                          BIT10== 2000
001000                          BIT09== 1000
000400                          BIT08== 400
000200                          BIT07== 200
000100                          BIT06== 100
000040                          BIT05== 40
000020                          BIT04== 20
000010                          BIT03== 10
000004                          BIT02== 4
000002                          BIT01== 2
000001                          BIT00== 1
                                ;
001000                          BIT9==  BIT09
000400                          BIT8==  BIT08
000200                          BIT7==  BIT07
000100                          BIT6==  BIT06
000040                          BIT5==  BIT05
000020                          BIT4==  BIT04
000010                          BIT3==  BIT03
000004                          BIT2==  BIT02
000002                          BIT1==  BIT01
000001                          BIT0==  BIT00
                                ;
                                ; EVENT FLAG DEFINITIONS
                                ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                ;
000040                          EF.START==      32.                     ; START COMMAND WAS ISSUED
000037                          EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
000036                          EF.CONTINUE==   30.                     ; CONTINUE COMMAND WAS ISSUED
000035                          EF.NEW==        29.                     ; A NEW PASS HAS BEEN STARTED
000034                          EF.PWR==        28.                     ; A POWER-FAIL/POWER-UP OCCURRED
                                ;
                                ;
                                ; PRIORITY LEVEL DEFINITIONS
                                ;
000340                          PRI07== 340
000300                          PRI06== 300
```

```
                000240                  PRI05== 240
                000200                  PRI04== 200
                000140                  PRI03== 140
                000100                  PRI02== 100
                000040                  PRI01== 40
                000000                  PRI00== 0
                                        ;
                                        ;OPERATOR FLAG BITS
                                        ;
                000004                  EVL==       4
                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
  2199
  2207
  2208                                  ; REGISTER USAGE.
  2209                                  ;
  2210                                  ;       R0 - PASSES PARAMETERS TO/FROM DIAGNOSTIC SUPERVISOR.
  2211                                  ;       R1 - COMMAND SEQUENCE TABLE POINTER.
  2212                                  ;       R2 - GENERAL PURPOSE REGISTER.
  2213                                  ;       R3 - GENERAL PURPOSE REGISTER.
  2214                                  ;       R4 - GENERAL PURPOSE REGISTER.
  2215                                  ;       R5 - CURRENT LOGICAL DEVICE NUMBER X 2.
  2216                                  ;       R6 - STACK POINTER.
  2217                                  ;       R7 - PROGRAM COUNTER.
  2218
  2219                                  ;THE FOLLOWING ARE BIT DEFINITIONS FOR THE TSSR REGISTERS.
  2220
  2221          100000                  TS.SC==100000                   ;SPECIAL CONDITION BIT.
  2222          040000                  TS.UPE==40000                   ;UNIBUS PARITY ERROR
  2223          020000                  TS.SPE==20000                   ;SERIAL BUS PARITY ERROR.
  2224          010000                  TS.RMR==10000                   ;REGISTER MODIFICATION REFUSED.
  2225          004000                  TS.NXM==4000                    ;NON-EXISTENT MEMORY.
  2226          002000                  TS.NBA==2000                    ;NEED BUFFER ADDRESS.
  2227          001000                  TS.A17==1000                    ;BUS ADDRESS BIT 17.
  2228          000400                  TS.A16==400                     ;BUS ADDRESS BIT 16.
  2229          000200                  TS.SSR==200                     ;UNIT READY BIT.
  2230          000100                  TS.OFL==100                     ;OFF LINE.
  2231          177717                  TSC.FCC==177717                 ;FATAL CLASS CODE MASK.
  2232          177761                  TSC.TCC==177761                 ;TERMINATION CLASS CODE MASK.
```

```
                        ;THE FOLLOWING ARE BIT DEFINITIONS FOR THE COMMAND WORD

100000                  ACK.C==100000           ;ACKNOWLEDGE BIT
040000                  CVC.C==40000            ;CLEAR VOLUME CHECK.
020000                  OPP.C==20000            ;OPPOSITE BIT
010000                  SWB.C==10000            ;SWAP BYTE BIT
004000                  MOD.C3==4000            ;MODE BIT 3
004000                  BRF.C==4000             ;BYTE/RECORD/FILE COUNT FLAG BIT.  NOT USED
                                                ;BY TK25 BUT USED INTERNALLY BY THIS PROGRAM ONLY.
002000                  MOD.C2==2000            ;MODE BIT 2
001000                  MOD.C1==1000            ;MODE BIT 1
000400                  MOD.C0==400             ;MODE BIT 0
000200                  IE.C==200               ;INTERRUPT ENABLE
000100                  FMT.C1==100             ;FORMAT BIT 1
000100                  VFY.C==100              ;WRITE VERIFY FLAG BIT.  INTERNAL USE ONLY.
                                                ;NOT USED BY TK25.
000040                  FMT.C0==40              ;FORMAT BIT 0.
000040                  JMP.C==40               ;JUMP BIT-TO DIRECT THIS PROGRAM TO JUMP TO
                                                ;A CERTAIN LOCATION IN THE COMMAND SEQUENCE
                                                ;TABLE.  INTERNAL USE ONLY.
000020                  CMD.C4==20              ;COMMAND BIT 4
000020                  DLY.C==20               ;INSERT DELAY. INTERNAL USE ONLY.
000010                  CMD.C3==10              ;COMMAND BIT 3
000004                  CMD.C2==4               ;COMMAND BIT 2
000002                  CMD.C1==2               ;COMMAND BIT 1
000001                  CMD.C0==1               ;COMMAND BIT 0

                        ;       BIT DEFINITIONS FOR DEVICE CHARACTERISTICS.

000200                  CH.ESS==200             ;ENABLE SKIP TAPE MARKS STOP (STOP AT LOGICAL EOT).
000040                  CH.EAI==40              ;ENABLE ATTENTION INTERRUPTS.
000020                  CH.ERI==20              ;ENABLE MESSAGE BUFFER RELEASE INTERRUPTS.
000040                  DFTSCH==CH.EAI          ;DEFAULT CHARACTERISTICS CODE.

                        ;THE FOLLOWING INDICATES THE RELATIVE POSITIONS OF THE STATUS WORDS
                        ;IN THE MESSAGE BUFFER.

000004                  MS.RFC==4               ;RESIDUAL FRAME COUNT.
000006                  MS.XS0==6               ;EXT STATUS REG 0
000010                  MS.XS1==10              ;EXT STATUS REG 1
000012                  MS.XS2==12              ;EXT STATUS REG 2
000014                  MS.XS3==14              ;EXT STATUS REG 3

                        ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 0.

100000                  X0.TMK==100000          ;TAPE MARK.
040000                  X0.RLS==40000           ;RECORD LENGTH SHORT.
020000                  X0.LET==20000           ;LOGICAL EOT.
010000                  X0.RLL==10000           ;RECORD LENGTH LONG.
000100                  X0.ONL==100             ;ON LINE BIT.
000002                  X0.BOT==2               ;BOT BIT.
000001                  X0.EOT==1               ;EOT BIT.

                        ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 1.

000010                  X1.EWN==BIT3
```

```
                                ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 2.

100000                          X2.OPM==100000                  ;OPERATION IN PROGRESS, TAPE MOVING

                                ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 3.

000010                          X3.DCK==10                      ;DENSITY CHECK.
157400                          X3.RNY==157400                  ;CAPSTAN RUNAWAY UDIAG ERROR CODE.

                                ;THE FOLLOWING DEFINITIONS SHOW THE RELATIVE POSITIONS OF THE COMMAND
                                ;PACKET ENTRIES.

000000                          CP.CMD==0                       ;CMDPKT+0==TK25 COMMAND.
000002                          CP.ADL==2                       ;CMDPKT+2==BUFFER ADDRESS LOW.
000004                          CP.ADH==4                       ;CMDPKT+4==BUFFER ADDRESS HIGH.
000006                          CP.CNT==6                       ;CKDPKT+6==BYTE/FILE/RECORD COUNT

                                ;       MISCELLANEOUS DEFINITIONS.

000340                          INTPRI==PRI07                   ;PRIORITY TO BE USED IN INTERRUPT STATE.
000010                          SCHCNT==10                      ;ARBITRARY BYTE LENGTH FOR CHARACTERISTIC
                                                                ;BUFFER LENGTH. (EVEN #)
000016                          MSGCNT==16                      ;MESSAGE BUFFER LENGTH IN BYTES. (EVEN #)
000020                          DIACNT==20                      ;DIAGNOSTIC COMMAND BUFFER EXTENT.
010000                          DATCNT==4096.                   ;MAXIMUM RECORD LENGTH IN BYTES,
                                                                ;THIS COUNT SHOULD BE A MULTIPLE OF 256 TO INSURE
                                                                ;PROPER READ/WRITE BUFFER ALLOCATION BY THE SUPER.
177740                          RNOPSC==177740                  ;RANDOM # OF OPERATIONS MASK.
000007                          RANP==7                         ;CODE TO SELECT RANDOM PATTERN.
000020                          RRECL==16.                      ;READ RECOVERY ATTEMPT LIMIT.
000020                          WRECL==16.                      ;WRITE RECOVERY ATTEMPT LIMIT.
153624                          RANBC==153624                   ;CONSTANT USED TO RESET RANDOM # GENERATOR BASE.
032561                          RANSC==32561                    ;CONSTANT USED TO RESET RANDOM # SAVE LOCATION.
177774                          NINUSE==177774                  ;NOT IN USE CODE FOR DEVICE STATE TABLE.
177740                          NCMD.C==ACK.C!CVC.C!OPP.C!SWB.C!MOD.C3!MOD.C2!MOD.C1!MOD.C0!IE.C!FMT.C1!FMT.C0
                                                                ;NOT "COMMAND" BITS.

                                ;THE FOLLOWING DEFINES THE COMMAND WORD FOR EACH TK25 COMMAND.

100013                          DRI==   ACK.C!CMD.C3!CMD.C1!CMD.C0
                                                                ;DRIVE INIT.

104001                          RDF==   ACK.C!BRF.C!CMD.C0
                                                                ;READ FORWARD

104401                          RDR==   ACK.C!BRF.C!MOD.C0!CMD.C0
                                                                ;READ REVERSE

104005                          WRT==   ACK.C!BRF.C!CMD.C0!CMD.C2
                                                                ;WRITE COMMAND

104105                          WTV==   ACK.C!BRF.C!VFY.C!CMD.C0!CMD.C2
                                                                ;WRITE VERIFY

104010                          SRF==   ACK.C!BRF.C!CMD.C3
                                                                ;SPACE RECORD FORWARD
```

```
2348          104410          SRR==   ACK.C!BRF.C!MOD.C0!CMD.C3
2349                                                  ;SPACE RECORD REVERSE
2350
2351          105401          RNR==   ACK.C!BRF.C!MOD.C1!MOD.C0!CMD.C0
2352                                  ;READ REV RETRY1 - REREAD NEXT REVERSE, IE. SPACE FWD, READ REVERSE
2353
2354          125401          RNF==   ACK.C!BRF.C!OPP.C!MOD.C1!MOD.C0!CMD.C0
2355                                  ;READ REV RETRY2 - REREAD NEXT FORWARD, IE.READ FORWARD, SPACE REVERSE
2356
2357          105001          RPF==   ACK.C!BRF.C!MOD.C1!CMD.C0
2358                                  ;READ FWD RETRY1 - REREAD PREVIOUS FORWARD, IE. SPACE REVERSE, READ FORWARD
2359
2360          125001          RPR==   ACK.C!BRF.C!OPP.C!MOD.C1!CMD.C0
2361                                  ;READ FWD RETRY2 - REREAD PREVIOUS REVERSE, IE. READ REVERSE, SPACE FORWARD
2362
2363          105005          WRR==   ACK.C!MOD.C1!BRF.C!CMD.C2!CMD.C0
2364                                                  ;WRITE RETRY
2365
2366          102010          RWD==   ACK.C!MOD.C2!CMD.C3
2367                                                  ;REWIND COMMAND
2368
2369          100012          MBR==   ACK.C!CMD.C3!CMD.C1
2370                                                  ;MESSAGE BUFFER RELEASE
2371
2372          100011          WTM==   ACK.C!CMD.C3!CMD.C0
2373                                                  ;WRITE TAPE MARK.
2374
2375          101011          WTR==   ACK.C!MOD.C1!CMD.C3!CMD.C0
2376                                                  ;WRITE TAPE MARK RETRY.
2377
2378          105010          SFF==   ACK.C!BRF.C!MOD.C1!CMD.C3
2379                                                  ;SPACE FILE FORWARD
2380
2381          105410          SFR==   ACK.C!BRF.C!MOD.C0!MOD.C1!CMD.C3
2382                                                  ;SPACE FILE REVERSE
2383
2384          100017          GES==   ACK.C!CMD.C0!CMD.C1!CMD.C2!CMD.C3
2385                                                  ;GET EXTENDED STATUS
2386
2387          100411          ERS==   ACK.C!MOD.C0!CMD.C3!CMD.C0
2388                                                  ;ERASE 3 INCHES OF TAPE
2389
2390          100412          UNL==   ACK.C!MOD.C0!CMD.C3!CMD.C1
2391                                                  ;UNLOAD COMMAND
2392
2393          101012          CLN==   ACK.C!MOD.C1!CMD.C3!CMD.C1
2394                                                  ;ERASE TAPE.
2395
2396          140004          SCH==   ACK.C!CVC.C!CMD.C2      ;SET DEVICE CHARACTERISTICS.
2397
2398          100006          DIA==   ACK.C!CMD.C2!CMD.C1     ;DIAGNOSTICS.
2399
2400          000040          JMP==   JMP.C                   ;JUMP TO "N"TH COMMAND
2401
2402          000020          DLY==   DLY.C                   ;DELAY "N" MS.
2403
2404          177777          END==   177777                  ;END OF COMMAND SEQUENCES
```

```
                                .SBTTL  GLOBAL DATA SECTION

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

                                ;       COMMAND PACKET.

002314                          .       =       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002314  000000                  CMDPKT:: 0                      ;1ST WORD IS TK25 COMMAND.
002316  000000                          0                       ;2ND WORD IS THE BUFFER LOW ADDRESS.
002320  000000                          0                       ;3RD WORD IS THE BUFFER HIGH ADDRESS.
002322  000000                          0                       ;4TH WORD IS THE BYTE/RECORD/FILE COUNT.

                                ;       GET STATUS COMMAND PACKET.

002324                          .       =       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002324  100017                  GSCPK:: .WORD   GES

                                ;       MESSAGE BUFFER RELEASE COMMAND PACKET.

002330                          .       =       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002330  100012                  BRCPK:: .WORD   MBR

                                ;       REWIND COMMAND PACKET (USED IN ERROR RECOVERY ONLY)

002334                          .       =       .+3&177774      ;MUST BE ON A MODULE 4 BOUNDARY.
002334  102010                  RWCPK:: .WORD   RWD
002336  000001                          .WORD   1

                                ;       WORK AREA FOR ANALYSIS OF MESSAGE PACKET CONTENTS.

002340                          MSGPKT:: .BLKW  7               ;1ST WORD:: MESSAGE TYPE.
                                                                ;2ND WORD:: DATA FIELD LENGTH.
                                                                ;3RD WORD:: RESIDUAL FRAME COUNT.
                                                                ;4TH WORD:: XSTAT0
                                                                ;5TH WORD:: XSTAT1
                                                                ;6TH WORD:: XSTAT2
                                                                ;7TH WORD:: XSTAT3

                                ;       MESSAGE PACKETS.

002356                          MSGPK0:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #0
002374                          MSGPK1:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #1
002412                          MSGPK2:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #2
002430                          MSGPK3:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #3
```

```
                                ;       SET CHARACTERISTIC BLOCK.

002446  002356                  SCHBK:: MSGPK0                  ;1ST WORD:: MSGPKT ADDR LO(SET UP BY EXCUTE ROUTINE).
002450  000000                          0                       ;2ND WORD:: MSGPKT ADDR HI.
002452  000016                          MSGCNT                  ;3RD WORD:: MSG BUFFER LENGTH (BYTES)
002454  000040                          CH.EAI                  ;4TH WORD:: CHARACTERISTICS WORD(SET BY SETUP ROUTINE).

                                ;       TK25 REGISTER ADDRESSES.

002456                          TSDB:: .BLKW    4               ;TK25 DATA BUFFER ADDRESSES.
002466                          TSSR:: .BLKW    4               ;TK25 STATUS REGISTER ADDRESSES.
002476                          TSVCT:: .BLKW   4               ;TK25 VECTOR ADDRESSES.
        002456                  TSBA==TSDB                      ;DATA BUFFER ADDRESS REGISTER.

                                ;       ADDRESSES OF MESSAGE PACKETS.

002506  002356                  MSGPKA:: MSGPK0                 ;DEVICE 0.
002510  002374                           MSGPK1                 ;DEVICE 1.
002512  002412                           MSGPK2                 ;DEVICE 2.
002514  002430                           MSGPK3                 ;DEVICE 3.

                                ;       ADDRESSES OF INTERRUPT HANDLING ROUTINES.

002516  006564                  TS4INT:: TS4IN0                 ;DEVICE 0.
002520  006606                           TS4IN1                 ;DEVICE 1.
002522  006630                           TS4IN2                 ;DEVICE 2.
002524  006652                           TS4IN3                 ;DEVICE 3.

                                ;       TK25 CODE LEVELS, WILL BE STORED AFTER SCH CMD IN BASIC FUNCTION TEST

002526  000000                  TS4CL:: 0                       ;DEVICE 0
002530  000000                          0                       ;DEVICE 1
002532  000000                          0                       ;DEVICE 2
002534  000000                          0                       ;DEVICE 3

                                ;       UNIT NUMBERS OF ALL DEVICES BEING TESTED(1-4).
                                ;       WHEN DEVICE IS NOT IN USE, IT,S LOCATION WILL = -3.
                                ;       R5 WILL ALWAYS CONTAIN THE PRESENT LOGICAL UNIT NUMBER X 2.

002536  177774                  DEVTBL:: .WORD  NINUSE
002540  177774                           .WORD  NINUSE
002542  177774                           .WORD  NINUSE
002544  177774                           .WORD  NINUSE
002546  177777                           .WORD  END

                                ;       BAD TAPE TABLE POINTER: USED BY WRITE RETRY ROUTINE
                                ;       "WRTY" TO LOG BAD TAPE SPOTS ON UNITS UNDER TEST

002550  003000                  BTADDR:: BT0
002552  003052                           BT1
002554  003124                           BT2
002556  003176                           BT3
```

```
                                ;       COUNTER AREA.
                                ;
                        002560  CNTBGN=.
002560                          WRBC::  .BLKW   20              ;BYTES WRITTEN.
002620                          RRBC::  .BLKW   20              ;BYTES READ REV.
002660                          RFBC::  .BLKW   20              ;BYTES READ FWD.
002720                          WRREC:: .BLKW   4               ;RECOVERABLE WRITE ERRORS.
002730                          WRUNR:: .BLKW   4               ;UNRECOVERABLE WRITE ERRORS.
002740                          RRREC:: .BLKW   4               ;RECOVERABLE READ REV ERRORS.
002750                          RRUNR:: .BLKW   4               ;UNRECOVERABLE READ REV ERRORS.
002760                          RFREC:: .BLKW   4               ;RECOVERABLE READ FWDERRORS.
002770                          RFUNR:: .BLKW   4               ;UNRECOVERABLE READ FWD ERRORS.
003000                          BT0::   .BLKW   21.             ;UNIT 0 BAT TAPE SPOTS LOG
003052                          BT1::   .BLKW   21.             ;UNIT 1 BAT TAPE SPOTS LOG
003124                          BT2::   .BLKW   21.             ;UNIT 2 BAT TAPE SPOTS LOG
003176                          BT3::   .BLKW   21.             ;UNIT 3 BAT TAPE SPOTS LOG
003250                          WRTYCT:: .BLKW  4               ;WRITE RETRY COUNTER
003260                          PASCNT:: .BLKW  4               ;PASS COUNT.
003270                          SCCNT:: .BLKW   4               ;SPECIAL CONDITION COUNT.
003300                          VFYCNT:: .BLKW  4               ;COUNT OF TK25 DATA COMPARE ERRORS.
003310                          HRDCNT:: .BLKW  4               ;COUNT OF HARD ERRORS.
003320                          FTLCNT:: .BLKW  4               ;COUNT OF FATAL ERRORS.
        003330                  CNTEND=.                        ;END OF STATICTICAL COUNTERS.
003330                          RECCNT:: .BLKW  4               ;NUMBER OF RECORDS FROM BOT: CLEARED  ON REWIND
                                                                ;AND WHEN RESTARTING OR CONTINUING TEST 2.
        000550                  CNTLEN==CNTEND-CNTBGN           ;LENGTH OF STATISTICAL COUNTER AREA.

003340  000                     HERE:   .BYTE 0                 ;TEST 3 ASCII SEMAPHORE
                                ;       THE FOLLOWING ARE THE DEFINITIONS OF VARIABLES
                                ;       USED BY THE PROGRAM.

                                .EVEN
003342  000000                  RAMHLD:   .WORD  0              ;RAM ADDR HOLDER 1ST ADDRESS
003344  000000                  RAMR5H:   .WORD  0              ;HOLDS R5 FOR LATER
003346                          RAMDATA::.BLKW  16.             ;DATA READ FROM RAM PACKET OR MESSAGE BUF AREA
003406  000000                  RAMSIZ::  .WORD  0              ;RAM DATA SIZE FOR PRAMPKT ROUTINE
003410  000000                  CMPDAT::  .WORD  0              ;COUNTS # OF READS (TEST 3) BEFORE ALLOWING A DATA COMPARE
003412  000003                  DATRAT::  .WORD  3              ;CONTROLS THE DATA COMPARE RATIO
003414  000000                  DATAWT::  .WORD  0              ;WRITE BUFFER ADDRESS.
003416  000000                  DATARD::  .WORD  0              ;READ BUFFER ADDRESS.
003420  000000                  NCNT::    .WORD  0              ;STORAGE FOR VALUE OF N.
003422  000000                  NCNT1::   .WORD  0              ;TEMP STORAGE FOR VALUE OF N.
003424  000000                  BRFCNT::  .WORD  0              ;STORAGE FOR BPCR VALUE.
003426  177777                  CMDWRD::  .WORD  END            ;CONTAINS COMMAND WORD BEING EXECUTED PRESENTLY.
003430  177777                  CMDSAV::  .WORD  END            ;SAVE LOCATION FOR CMD WORD DURING ERROR RECOVERY
003432  177777                  PCMDWD::  .WORD  END            ;CONTAINS PREVOUS COMMAND WORD.
003434  000000                  CMDLG::   .WORD  0              ;CURRENT COMMAND LOGGING CODE.
003436  000000                  LENMSK::  .WORD  0              ;RANDOM WRITE LENGTH MASK, TO BE SET UP BY TESTS
003440  153624                  RANB::    .WORD  153624         ;RANDOM # GENERATOR BASE.
003442  032561                  RANS::    .WORD  32561          ;RANDOM # SAVE LOCATION.
003444  000000                  TIME1::   .WORD  0              ;TIME COUNT 1.
003446  000000                  TIME2::   .WORD  0              ;TIME COUNT 2.
003450  000000                  JLOOP::   .WORD  0              ;JMP COMMAND LOOP COUNT.
003452  000000                  JLOC::    .WORD  0              ;JMP COMMAND LOCATION COUNT.
003454  000000                  PATERN::  .WORD  0              ;PATTERN SELECT CODE.
```

GLOBAL DATA SECTION

SEQ 0056

```
                                CTCC::    .WORD  0              ;CURRENT TERMINATION CLASS CODE.
                                R5SAVE::  .WORD  0              ;LOCATION FOR SAVING CURRENT DEVICE POINTER.
                                TSSREG::  .WORD  0              ;CURRENT STATUS REGISTER.
                                DIABLK==DATAWT                  ;WRITE BUFFER ALSO USED FOR DIAG CMD.
                                LCSR::    .WORD  0              ;L CLOCK CSR ADDRESS
                                LBRVEC::  .WORD  0              ;L CLOCK BR LEVEL
                                LVECT::   .WORD  0              ;L CLOCK VECTOR ADDRESS
                                LHERTZ::  .WORD  0              ;L CLOCK FREQUENCY

                                ;       ERROR FLAG AREA, THESE FLAGS ARE CLEARED DURING INITIALIZATION AND
                                ;       AFTER EACH COMMAND IS COMPLETED.

                        003474  BGNFLG=.
                        003474  RETRYC::  .WORD  0              ;# OF RECOVERY ATTEMPTS EXECUTED.
                        003476  RPTCNT::  .BYTE  0      ;WRITE REPEAT ON SAME SPOT CNTR: 4 PER WRITE RETRY
                        003477  WRTYFG::  .BYTE  0              ;WRITE RETRY ON SAME SPOT IN PROGRESS FLAG
                        003500  WRTYER::  .BYTE  0              ;WRITE RETRY ON SAME SPOT ERROR FLAG
                        003501  RECLOG::  .BYTE  0              ;RECORD COUNT HAS BEEN UPDATED FOR THIS RECORD.
                        003502  ERLOG::   .BYTE  0              ;DATA BYTES AND ERRORS HAVE BEEN LOGGED FOR THIS RECORD.
                        003503  RWERR::   .BYTE  0              ;READ/WRITE ERROR HAS OCCURED.
                        003504  UNREC::   .BYTE  0              ;UNRECOVERABLE ERROR HAS OCCURED.
                        003505  ERRREC::  .BYTE  0              ;ERROR RECOVERY MODE.
                                          .EVEN
                        003506  ENDERF=.

                                ;       ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED DURING INITIALIZATION.

                        003506  INTFLG::  .BLKW  4              ;INTERRUPT OCCURRED FLAGS FOR EACH DEVICE.
                        003516  EOTFLG::  .BLKW  4              ;EOT/BOT FLAGS FOR EACH DEVICE (XSTAT0).
                        003526  BTPT::    .WORD  0              ;BAD TAPE SPOT POINTER TO BT0-BT3 VIA BTADDR
                        003530  TC4STO::  .WORD  0              ;TCC4 ROUTINE - TEMPORARY STORAGE
                        003532  EXPBOT::  .BYTE  0              ;BOT IS EXPECTED, DO NOT ABORT ON BOT/FUNC RTI.
                        003533  RANDOM::  .BYTE  0              ;RANDOM EVERYTHING FLAG.
                        003534  VFYFLG::  .BYTE  0              ;SET DURING WRITE/VERIFY COMMAND.
                        003535  RPTFLG::  .BYTE  0              ;PERFORMANCE REPORT HAS BEEN REQUESTED.
                        003536  SWBFLG::  .BYTE  0              ;ENABLES SWAP BYTE FUNCTION WHEN NOT EQUAL TO ZERO.
                        003537  LOOPCT::.BYTE    0              ;WRITE LOOP CONTROL (TEST 3)
                        003540  IRE::     .BYTE  0              ;INHIBIT RESIDUAL FRAME COUNT ERROR REPORT.
                        003541  DROPED::  .BYTE  0              ;CURRENT UNIT HAS BEEN DROPPED
                        003542  T1SWB::   .BYTE  0              ;TEST1 SWAP BYTES FLAG
                        003543  ALLEOT::  .BYTE  0              ;ALL UNITS  @ EOT FLAG
                        003544  STREAM::  .BYTE  0              ;INDICATES TEST ONE UNIT AT A TIME, COMPLETELY.
                        003545  ERSFLG::  .BYTE  0              ;ERASE FLAG:  DO ERASE AFTER A SPACE REV TO DELETE
                                                                ;BADLY WRITTEN RECORD.  1 TO 4 ERASES LEAVING
                                                                ;A 3 TO 12 INCH GAP MAY RESULT.

                                          .EVEN
                        003546  ENDFLG=.

                                ;       ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED ONLY AFTER BEING CHECKED.

                        003546  STAFLG::  .BYTE  0              ;START FLAG - SET BY INIT CODE IF STARTING.
                        003547  PWRFLG::  .BYTE  0              ;POWER FAILURE FLAG - SET ONLY DURING INIT.
                        003550  TRAPD4::  .BYTE  0              ;TRAPED AT 4 FLAG
                        003551  MISCFG::  .BYTE  0              ;MISCELLANEOUS FLAG
```

```
                                ;       OPERATOR FLAG SETTINGS PASSED BY DIAG. SUPERVISOR IN A 16 BIT WORD
                                ;       SEE GLOBAL EQUATES SECTION FOR FLAG BIT LIST

003552  000000                  OPFLAG:: .WORD  0       ;READ ONLY OPERATOR FLAG WORD
                                        .EVEN

                                        ;THE FOLLOWING IS THE COMMAND SEQUENCE TABLE.  THE TABLE
                                        ;HAS DEFAULT VALUES AT PROGRAM LOAD AS SHOWN.  THESE VALUES
                                        ;CAN BE UPDATED BY A TEST OR BY OPERATOR INPUT.

003554  140004                  CMDSEQ:: .WORD  SCH             ;SET CHARACTERISTICS.
003556  000040                          .WORD   CH.EAI
003560  000001                          .WORD   1
003562  000000                          .WORD   0
003564  102010                  CMDSE2:: .WORD  RWD             ;REWIND.
003566  000001                          .WORD   1               ;BYTE COUNT.
003570  000001                          .WORD   1               ;ONCE.
003572  000007                          .WORD   RANP            ;PATTERN.
003574  104005                          .WORD   WRT             ;WRITE.
003576  010000                          .WORD   DATCNT          ;MAX BUFFER LENGTH.
003600  035230                          .WORD   15000.          ;15000 RECORDS.
003602  000007                          .WORD   RANP            ;PATTERN - RANDOM DATA
003604  102010                          .WORD   RWD             ;REWIND
003606  000001                          .WORD   1               ;BYTE COUNT
003610  000001                          .WORD   1               ;ONE ITERATION
003612  000007                          .WORD   RANP            ;RANDOM DATA
003614  104001                          .WORD   RDF             ;READ FWD.
003616  010000                          .WORD   DATCNT          ;MAX BUFFER LENGTH.
003620  035230                          .WORD   15000.          ;15000 RECORDS.
003622  000007                          .WORD   RANP            ;RANDOM DATA
003624  102010                          .WORD   RWD             ;REWIND
003626  000001                          .WORD   1               ;BYTE COUNT.
003630  000001                          .WORD   1               ;ONCE.
003632  000007                          .WORD   RANP            ;PATTERN.
003634                                  .BLKW   4               ;EXTENSION TO HOLD 1 MORE CMD.
003644  177777                  SEQEND:: .WORD  END             ;SOFT END OF SEQUENCE TABLE.
003646  177777                          .WORD   END
003650  177777                          .WORD   END
003652  177777                          .WORD   END
003654  177777                          .WORD   END             ;HARD END OF SEQUENCE TABLE.
```

```
                                        ;THE FOLLOWING IS THE TK25 COMMAND TABLE

003656  100013  CMDTBL:: .WORD  DRI      ;DRIVE INIT.
003660  104001          .WORD   RDF      ;READ FORWARD.
003662  104401          .WORD   RDR      ;READ REVERSE.
003664  104005          .WORD   WRT      ;WRITE
003666  104105          .WORD   WTV      ;WRITE/VERIFY. (WRITE ALL RECORDS, RDR AND
                                         ;CHECK DATA ON ALL RECORDS, RDF AND
                                         ;CHECK DATA ON ALL RECORDS.)
003670  104010          .WORD   SRF      ;SPACE "N" RECORDS FORWARD.
003672  104410          .WORD   SRR      ;SPACE "N" RECORDS REVERSE.
003674  105401          .WORD   RNR      ;READ NEXT REVERSE. I.E., SPACE FWD, READ REVERSE.
003676  125401          .WORD   RNF      ;READ NEXT FORWARD, I.E., READ FORWARD, SPACE REVERSE.
003700  105001          .WORD   RPF      ;READ PREVIOUS FORWARD. I.E., SPACE REVERSE, READ FORWARD
003702  125001          .WORD   RPR      ;READ PREVIOUS REVERSE. I.E., READ REVERSE, SPACE FORWARD
003704  105005          .WORD   WRR      ;WRITE RETRY.
003706  102010          .WORD   RWD      ;REWIND.
003710  100012          .WORD   MBR      ;MESSAGE BUFFER RELEASE
003712  100011          .WORD   WTM      ;WRITE TAPE MARK
003714  101011          .WORD   WTR      ;WRITE TAPE MARK RETRY.
003716  105010          .WORD   SFF      ;SPACE "N" FILES FORWARD.
003720  105410          .WORD   SFR      ;SPACE "N" FILES REVERSE.
003722  100017          .WORD   GES      ;GET EXTENDED STATUS.
003724  100411          .WORD   ERS      ;ERASE 3 INCHES OF TAPE.
003726  100412          .WORD   UNL      ;REWIND AND UNLOAD.
003730  101012          .WORD   CLN      ;CLEAR TAPE.
003732  140004          .WORD   SCH      ;SET CHARACTERISTICS.
003734  100006          .WORD   DIA      ;DIAGNOSTIC COMMAND.
003736  000040          .WORD   JMP      ;JUMP TO THE NTH COMMAND IN THE SEQUENCE.
003740  000020          .WORD   DLY      ;DELAY "N" MS.
003742  177777          .WORD   END      ;END OF COMMAND TABLE
```

```
                                        ;       THE FOLLOWING TABLE CONTAINS THE ASCII FOR EACH COMMAND.
                                        ;
003744  104     122     111     CMDASC:: .ASCII /DRI/           ;DRIVE INIT.
003747  122     104     106             .ASCII  /RDF/           ;READ FORWARD.
003752  122     104     122             .ASCII  /RDR/           ;READ REVERSE.
003755  127     122     124             .ASCII  /WRT/           ;WRITE
003760  127     124     126             .ASCII  /WTV/           ;WRITE/VERIFY.  (WRITE ALL RECORDS, RDR AND CHECK DATA
                                                                ;ON ALL RECORDS, RDF AND CHECK DATA ON ALL RECORDS.)
003763  123     122     106             .ASCII  /SRF/           ;SPACE "N" RECORDS FORWARD.
003766  123     122     122             .ASCII  /SRR/           ;SPACE "N"  RECORDS REVERSE.
003771  122     116     122             .ASCII  /RNR/           ;READ NEXT REVERSE. I.E., SPACE FWD READ REVERSE.
003774  122     116     106             .ASCII  /RNF/           ;READ NEXT FORWARD, I.E., READ FORWARD, SPACE REVERSE.
003777  122     120     106             .ASCII  /RPF/           ;READ PREVIOUS FORWARD. IE., SPACE REVERSE, READ FORWARD
004002  122     120     122             .ASCII  /RPR/           ;READ PREVIOUS REVERSE. IE., READ REVERSE, SPACE FORWARD
004005  127     122     122             .ASCII  /WRR/           ;WRITE RETRY.
004010  122     127     104             .ASCII  /RWD/           ;REWIND.
004013  115     102     122             .ASCII  /MBR/           ;MESSAGE BUFFER RELEASE
004016  127     124     115             .ASCII  /WTM/           ;WRITE TAPE MARK
004021  127     124     122             .ASCII  /WTR/           ;WRITE TAPE MARK RETRY.
004024  123     106     106             .ASCII  /SFF/           ;SPACE "N" FILES FORWARD.
004027  123     106     122             .ASCII  /SFR/           ;SPACE "N" FILES REVERSE.
004032  107     105     123             .ASCII  /GES/           ;GET EXTENDED STATUS.
004035  105     122     123             .ASCII  /ERS/           ;ERASE 3 INCHES OF TAPE.
004040  125     116     114             .ASCII  /UNL/           ;REWIND AND UNLOAD.
004043  103     114     116             .ASCII  /CLN/           ;CLEAN TAPE.
004046  123     103     110             .ASCII  /SCH/           ;SET CHARACTERISTICS.  WHERE BRF=200, 40, 20, 0.
                                                                ;SEE TK25 PROGRAMMING SPECIFICATION FOR DESCRIPTION.
004051  104     111     101             .ASCII  /DIA/           ;DIAGNOSTICS.  SEE TK25 PROGRAMMING SPECIFICATION
                                                                ;FOR DESCRIPTION.  ODT MUST BE USED TO LOAD DIAGNOSTIC DATA
                                                                ;INTO THE WRITE BUFFER BEFORE THIS CMD IS ISSUED.
004054  112     115     120             .ASCII  /JMP/           ;JUMP TO THE NTH COMMAND IN THE COMMAND
                                                                ;SEQUENCE TABLE, WHERE N IS DEFINED IN
                                                                ;THE # OF OPERATIONS.
004057  104     114     131             .ASCII  /DLY/           ;DELAY "N" MS, WHERE N IS DEFINED IN
                                                                ;THE # OF OPERATIONS.
004062  105     116     104             .ASCII  /END/           ;END OF COMMAND SEQUENCE.
                                        .EVEN
```

```
                                .SBTTL  GLOBAL TEXT SECTION

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


                                ;
                                ; FORMAT STATEMENTS USED IN PRINT CALLS
                                ;
                                        .NLIST  BEX
004066  045  116  045  CODELM:: .ASCIZ /%N%AUNIT %D1%A TK25 CODE LEVEL P%03%N%N/
                                .EVEN
004136  130  130  130  HALTM:: .ASCIZ  /XXX CMD - TYPE <CR> TO CONTINUE/
004176  103  115  104  CMDPKM:: .ASCIZ /CMD PACKET ADR NOT ON MODULO 4 BOUNDARY: RELOAD!/
                                .EVEN
004260  104  101  124  WTVERM:: .ASCIZ /DATA COMPARE ERROR/
004303  116  117  040  TOERM:: .ASCIZ  /NO TK25 RESPONSE/
004324  125  116  104  SCERM:: .ASCIZ  /UNDEFINED SPEC COND/
004350  122  106  103  RFCERM:: .ASCIZ /RFC NON ZERO/
004365  124  113  062  NSSRM:: .ASCIZ  /TK25 NOT READY/
004404  122  105  124  RLEXM:: .ASCIZ  /RETRY LIMIT EXCEEDED/
004431  125  116  111  ATTNM:: .ASCIZ  /UNIT OFF LINE/
004447  106  125  116  FUNRM:: .ASCIZ  /FUNCTION REJECT/
004467  106  101  124  FATSM:: .ASCIZ  /FATAL SUBSYSTEM ERROR/
004515  116  117  040  NOINTM:: .ASCIZ /NO INTERRUPT/
004532  124  101  120  TSAM::   .ASCIZ  /TAPE STATUS ALERT/
004554  124  117  117  TOOMM:: .ASCIZ  /TOO MANY INTERRUPTS/
004600  103  101  120  RNYM::   .ASCIZ  /CAPSTAN RUNAWAY-GET STATUS RESULTS:/
004644  122  105  103  RERM::   .ASCIZ  /RECOVERABLE ERROR/
004666  125  116  122  URERM:: .ASCIZ  /UNRECOVERABLE ERROR/
004712  045  116  045  DROPDM:: .ASCIZ /%N%ADROPPED UNIT %D1%N/
004741  045  116  045  AUDRPM:: .ASCIZ /%N%AALL UNITS DROPPED%N%N/
004773  045  116  045  DTAER2:: .ASCIZ "%N%ABYTE:%D4%S2%AWAS:%B8%S2%AS/B:%B8%N"
005042  045  104  064  DTAER3:: .ASCIZ "%D4%A BYTES IN ERROR OUT OF %D4%N"
005104  045  101  116  DTAER4:: .ASCIZ /%ANO DATA READ%N/
005125  045  101  122  DTAER5:: .ASCIZ /%ARECORD TOO LONG: >%04%A BYTES%N/
005167  045  101  122  NURTY1:: .ASCIZ /%ARECOVERED ON RETRY #%D2%N/
005223  045  101  125  OFLINM:: .ASCIZ /%AUNIT %D1%A OFF LINE%N/
005253  045  101  107  GETSTM:: .ASCIZ /%AGET STATUS CMD RESULTS:%N/
005307  045  116  000  CRLF::   .ASCIZ  /%N/
005312  045  116  045  CRLFSP:: .ASCIZ /%N%S7/
005320  045  116  045  RAMFHR: .ASCIZ  '%N%A ********** CONTROLLER RAM DUMP **********'
005377  045  116  045  RAMIOP: .ASCIZ  '%N%A RAM ADDRESS (OCTAL) = %03%A - %03%N'
005450  045  101  040  RAMPD:  .ASCIZ  '%A %03%A '
005462  045  116  045  RAMLIN: .ASCIZ  '%N%N%N'
                                .LIST   BEX
                                .EVEN
```

```
                        .SBTTL  GLOBAL ERROR REPORT SECTION

                        ;++
                        ; THE GLOBAL ERROR REPORT SECTION CONTAINS THE PRINTB AND PRINTX CALLS
                        ; THAT ARE USED IN MORE THAN ONE TEST.  IT ALSO INCLUDES THE ASCII MESSAGES
                        ; THAT ARE USED BY THE PRINTB AND PRINTX CALLS..
                        ;--


005472                          BGNMSG  DTAERM
005472                  DTAERM::
                                PRINTB  #STAER1,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
005472  016546  003330                                          MOV     RECCNT(R5),-(SP)
005476  016546  003260                                          MOV     PASCNT(R5),-(SP)
005502  016546  002536                                          MOV     DEVTBL(R5),-(SP)
005506  012746  006152                                          MOV     #STAER1,-(SP)
005512  012746  000004                                          MOV     #4,-(SP)
005516  010600                                                  MOV     SP,R0
005520  104414                                                  TRAP    C$PNTB
005522  062706  000012                                          ADD     #12,SP
                                PRINTB  #STAER7
005526  012746  006244                                          MOV     #STAER7,-(SP)
005532  012746  000001                                          MOV     #1,-(SP)
005536  010600                                                  MOV     SP,R0
005540  104414                                                  TRAP    C$PNTB
005542  062706  000004                                          ADD     #4,SP
                                LET RECRED := R2                ;SAVE R2
005546  010237  006560                                          MOV     R2,RECRED
                                LET TIME1 := R3                 ;SAVE R3
005552  010337  003444                                          MOV     R3,TIME
                                LET TIME2 := R4                 ;SAVE R4
005556  010437  003446                                          MOV     R4,TIME2
005562  004737  006674          JSR PC,RECTAP
                                LET R2 := RECRED                ;RETRIEVE RECORD READ
005566  013702  006560                                          MOV     RECRED,R2
                                LET RECRED := R3                ;RESTORE R2
005572  010337  006560                                          MOV     R3,RECRED
                                LET R3 := TIME1                 ;SAVE RECORD READ
005576  013703  003444                                          MOV     TIME1,R3
                                LET R4 := TIME2                 ;RESTORE R3
005602  013704  003446                                          MOV     TIME2,R4
                                PRINTB  #STAER6,RECRED          ;RESTORE R4
005606  013746  006560                                          MOV     RECRED,-(SP)
005612  012746  006274                                          MOV     #STAER6,-(SP)
005616  012746  000002                                          MOV     #2,-(SP)
005622  010600                                                  MOV     SP,R0
005624  104414                                                  TRAP    C$PNTB
005626  062706  000006                                          ADD     #6,SP
                                EXIT    MSG
005632  000167                                                  .WORD   J$JMP
005634  000000                                                  .WORD   L10002-2-.

                                .EVEN

005636                          ENDMSG
                        L10002:
005636  104423                                                  TRAP    C$MSG
```

```
2826 005640                          BGNMSG  STAERM
     005640                    STAERM::
2827 005640                          PRINTB  #STAER1,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
     005640  016546  003330                                          MOV     RECCNT(R5),-(SP)
     005644  016546  003260                                          MOV     PASCNT(R5),-(SP)
     005650  016546  002536                                          MOV     DEVTBL(R5),-(SP)
     005654  012746  006152                                          MOV     #STAER1,-(SP)
     005660  012746  000004                                          MOV     #4,-(SP)
     005664  010600                                                  MOV     SP,R0
     005666  104414                                                  TRAP    C$PNTB
     005670  062706  000012                                          ADD     #12,SP
2828 005674                          PRINTB  #STAER7
     005674  012746  006244                                          MOV     #STAER7,-(SP)
     005700  012746  000001                                          MOV     #1,-(SP)
     005704  010600                                                  MOV     SP,R0
     005706  104414                                                  TRAP    C$PNTB
     005710  062706  000004                                          ADD     #4,SP
2829 005714                          LET R2 := CMDPKT CLR.BY #177740
     005714  013702  002314                                  MOV     CMDPKT,R2
     005720  042702  177740                                  BIC     #177740,R2
2830 005724                          LET R2 := R2 - #1
     005724  005302                                          DEC     R2
2831 005726                          IF R2 EQ #0 THEN                ;IF CMD IS A READ
     005726  005702                                          TST     R2
     005730  001016                                          BNE     50000$
2832 005732  004737  006674            JSR PC,RECTAP                 ;THEN RETRIEVE
2833 005736                            LET RECRED := R3              ;AND
     005736  010337  006560                                  MOV     R3,RECRED
2834 005742                            PRINTB #STAER6,RECRED         ;TYPE RECORD READ
     005742  013746  006560                                          MOV     RECRED,-(SP)
     005746  012746  006274                                          MOV     #STAER6,-(SP)
     005752  012746  000002                                          MOV     #2,-(SP)
     005756  010600                                                  MOV     SP,R0
     005760  104414                                                  TRAP    C$PNTB
     005762  062706  000006                                          ADD     #6,SP
2835 005766                          ENDIF
     005766                                                  50000$:
2836 005766                          PRINTX  #STAER2
     005766  012746  006330                                          MOV     #STAER2,-(SP)
     005772  012746  000001                                          MOV     #1,-(SP)
     005776  010600                                                  MOV     SP,R0
     006000  104415                                                  TRAP    C$PNTX
     006002  062706  000004                                          ADD     #4,SP
2837 006006                          PRINTX  #STAER3,CMDPKT,@TSDB(R5),MSGPKT+MS.RFC,TSSREG,CTCC
     006006  013746  003456                                          MOV     CTCC,-(SP)
     006012  013746  003462                                          MOV     TSSREG,-(SP)
     006016  013746  002344                                          MOV     MSGPKT+MS.RFC,-(SP)
     006022  017546  002456                                          MOV     @TSDB(R5),-(SP)
     006026  013746  002314                                          MOV     CMDPKT,-(SP)
     006032  012746  006407                                          MOV     #STAER3,-(SP)
     006036  012746  000006                                          MOV     #6,-(SP)
     006042  010600                                                  MOV     SP,R0
     006044  104415                                                  TRAP    C$PNTX
     006046  062706  000016                                          ADD     #16,SP
2838 006052                          PRINTX  #STAER4,CMDPKT+2,CMDPKT+4,CMDPKT+6
     006052  013746  002322                                          MOV     CMDPKT+6,-(SP)
     006056  013746  002320                                          MOV     CMDPKT+4,-(SP)
```

```
        006062  013746  002316                                                  MOV     CMDPKT+2,-(SP)
        006066  012746  006445                                                  MOV     #STAER4,-(SP)
        006072  012746  000004                                                  MOV     #4,-(SP)
        006076  010600                                                          MOV     SP,R0
        006100  104415                                                          TRAP    C$PNTX
        006102  062706  000012                                                  ADD     #12,SP
2839    006106                          PRINTX  #STAER5,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3
        006106  013746  002354                                                  MOV     MSGPKT+MS.XS3,-(SP)
        006112  013746  002352                                                  MOV     MSGPKT+MS.XS2,-(SP)
        006116  013746  002350                                                  MOV     MSGPKT+MS.XS1,-(SP)
        006122  013746  002346                                                  MOV     MSGPKT+MS.XS0,-(SP)
        006126  012746  006465                                                  MOV     #STAER5,-(SP)
        006132  012746  000005                                                  MOV     #5,-(SP)
        006136  010600                                                          MOV     SP,R0
        006140  104415                                                          TRAP    C$PNTX
        006142  062706  000014                                                  ADD     #14,SP
2840    006146                          EXIT    MSG
        006146  000167                                                          .WORD   J$JMP
        006150  000410                                                          .WORD   L10003-2-.
2841
2842                                    .NLIST  BEX
2843    006152      045     101     130 STAER1: .ASCIZ  /%AXXX CMD FAILED - UNIT %D1%S3%APASS:%D5%S3%ARECORD:%D5%N/
2844                                    .EVEN
2845    006244      045     101     120 STAER7: .ASCIZ  /%APREVIOUS CMD WAS  XXX/
2846    006274      045     123     061 STAER6: .ASCIZ  /%S11%A* RECORD READ:%D5%A */
2847    006330      045     116     045 STAER2: .ASCIZ  /%N%ACMDPKT%S2%ATSBA%S4%ARFC%S5%ATSSR%S3%ATCC%N/
2848    006407      045     117     066 STAER3: .ASCIZ  /%06%S2%06%S2%06%S2%06%S2%D1%N/
2849    006445      045     117     066 STAER4: .ASCII  /%06%N/
2850    006452      045     117     066         .ASCII  /%06%N/
2851    006457      045     117     066         .ASCIZ  /%06%N/
2852    006465      045     101     130 STAER5: .ASCII  /%AXST0%S4%4XST1%S4%AXST2%S4%AXST3%N/
2853    006530      045     117     066         .ASCIZ  /%06%S2%06%S2%06%S2%06%N/
2854                                    .LIST   BEX
2855                                    .EVEN
2856    006560  000000              RECRED: .WORD   0                   ;RECORD READ FROM TAPE
2857
2858    006562                          ENDMSG
        006562                  L10003:
        006562  104423                                                          TRAP    C$MSG
```

```
                        .SBTTL  GLOBAL SUBROUTINES SECTION
                        ;**
                        ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
                        ; THAT ARE USED IN MORE THAN ONE TEST.
                        ;--

                        ;       MODULES TO HANDLE TK25 INTERRUPTS.

                                BGNSRV  TS4IN0                  ;DEVICE 0.
                        TS4IN0::
                                LET INTFLG := INTFLG + #1       ;SET INTERRUPT OCCURRED FLAG.
                                                                        INC     INTFLG
                                ENDSRV  #0
                        L10004:
                                                                        BICB    #340,2(SP)
                                                                        BISB    #0,2(SP)
                                                                        RTI

                                BGNSRV  TS4IN1                  ;DEVICE 1.
                        TS4IN1::
                                LET INTFLG+2 := INTFLG+2 + #1   ;SET INTERRUPT OCCURRED FLAG.
                                                                        INC     INTFLG+2
                                ENDSRV  #0
                        L10005:
                                                                        BICB    #340,2(SP)
                                                                        BISB    #0,2(SP)
                                                                        RTI

                                BGNSRV  TS4IN2                  ;DEVICE 2.
                        TS4IN2::
                                LET INTFLG+4 := INTFLG+4 + #1   ;SET INTERRUPT OCCURRED FLAG.
                                                                        INC     INTFLG+4
                                ENDSRV  #0
                        L10006:
                                                                        BICB    #340,2(SP)
                                                                        BISB    #0,2(SP)
                                                                        RTI

                                BGNSRV  TS4IN3                  ;DEVICE 3.
                        TS4IN3::
                                LET INTFLG+6 := INTFLG+6 + #1   ;SET INTERRUPT OCCURRED FLAG.
                                                                        INC     INTFLG+6
                                ENDSRV  #0
                        L10007:
                                                                        BICB    #340,2(SP)
                                                                        BISB    #0,2(SP)
                                                                        RTI
```

```
2884
2885                                    ;       SUBROUTINE TO RETRIEVE RECORD COUNT READ FROM TAPE FOR ERROR
2886                                    ;       PRINTS.
2887                                    ;       INPUTS:
2888                                    ;       OUTPUTS:  R3 = RECORD COUNT READ
2889                                    ;       REGISTERS:  R2, R3, R4
2890                                    ;       CALLS:
2891
2892 006674                            RECTAP::IF @MOD.CO SETIN CMDWRD THEN    ;READ REV FETCH
       006674  032737  000400  003426                                                   BIT     @MOD.CO,CMDWRD
       006702  001430                                                                  BEQ     50001$
2893 006704                                 LET R2 := MSGPKT+MS.RFC + DATARD ;FIND LAST READ AD.
       006704  013702  002344                                                          MOV     MSGPKT+MS.RFC,R2
       006710  063702  003416                                                          ADD     DATARD,R2
2894 006714                                 IF @BIT00 SETIN R2 THEN         ;ODD AD., REASSEMBLE
       006714  032702  000001                                                          BIT     @BIT00,R2
       006720  001417                                                                  BEQ     50002$
2895 006722                                   LET R2 := R2 + #1             ;REC COUNT STARTING
       006722  005202                                                                  INC     R2
2896 006724                                   LET R3 :B= (R2) CLR.BY #177400  ;WITH UPPER BYTE FETCH
       006724  111203                                                                  MOVB    (R2),R3
       006726  142703  177400                                                          BICB    #177400,R3
2897 006732                                   LET R3 := SWAP R3             ;
       006732  000303                                                                  SWAB    R3
2898 006734                                   LET R2 := R2 - #1             ;LOWER BYTE AD.
       006734  005302                                                                  DEC     R2
2899 006736                                   IFB SWBFLG NE #0 THEN
       006736  105737  003536                                                          TSTB    SWBFLG
       006742  001401                                                                  BEQ     50003$
2900 006744                                     LET R2 := R2 - #1           ;LOWER BYTE AD. ON SWAP
       006744  005302                                                                  DEC     R2
2901 006746                                   ENDIF
       006746                                                                   50003$:
2902 006746                                   LET R4 :B= (R2) CLR.BY #177400 ;FETCH LOWER BYTE
       006746  111204                                                                  MOVB    (R2),R4
       006750  142704  177400                                                          BICB    #177400,R4
2903 006754                                     LET R3 := R3 OR R4            ;MERGE BYTES
       006754  050403                                                                  BIS     R4,R3
2904 006756                                   ELSE
       006756  000401                                                                  BR      50004$
       006760                                                                   50002$:
2905 006760                                     LET R3 := (R2)                ;EVEN AD. FETCH
       006760  011203                                                                  MOV     (R2),R3
2906 006762                                   ENDIF
       006762                                                                   50004$:
2907 006762                                 ELSE
       006762  000402                                                                  BR      50005$
       006764                                                                   50001$:
2908 006764                                   LET R3 := @DATARD             ;READ FWD FETCH
       006764  017703  174426                                                          MOV     @DATARD,R3
2909 006770                                 ENDIF
       006770                                                                   50005$:
2910
2911 006770  000207                    RTS     PC
```

```
                                ;       SUBROUTINE TO STORE A  SET CHARACTERISTIC COMMAND AS
                                ;       THE FIRST ENTRY IN THE SEQUENCE TABLE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

006772                          SETCH:: LET R1 := #CMDSEQ               ;INIT COMMAND SEQUENCE TABLE POINTER.
006772  012701  003554                                                          MOV     #CMDSEQ,R1
006776  012721  140004                  MOV     #SCH,(R1)+              ;THIS CODE SETS UP A SET CHARACTERISTIC
007002  012721  000040                  MOV     #DFTSCH,(R1)+           ;COMMAND AS THE FIRST COMMAND IN THE
007006  012721  000001                  MOV     #1,(R1)+                ;SEQUENCE TABLE.
007012  005721                          TST     (R1)+                   ;SKIP PATTERN LOCATION.
007014  000207                          RTS PC

                                ;       SUBROUTINE TO STORE A REWIND COMMAND IN THE SEQUENCE TABLE
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

007016                          SETRW:: LET (R1)+ := #RWD               ;CMD = REWIND.
007016  012721  102010                                                          MOV     #RWD,(R1)+
007022                                  LET (R1)+ := #1                 ;BRF.
007022  012721  000001                                                          MOV     #1,(R1)+
007026                                  LET (R1)+ := #1                 ;# OF OPERATIONS.
007026  012721  000001                                                          MOV     #1,(R1)+
007032  005721                          TST (R1)+                       ;SKIP PATTERN.
007034  000207                          RTS     PC                      ;RETURN
```

```
                                        ;       SUBROUTINE TO EXECUTE ALL COMMANDS IN THE SEQUENCE TABLE ON ALL
                                        ;       DEVICES.
                                        ;       INPUTS:
                                        ;       OUTPUTS:        R2 = TERMINATION INDICATOR (0=END OF TABLE,1=EOT)
                                        ;       REGISTERS:
                                        ;       CALLS:          CMDAC,SETUP,EXSUB,CKHAE,NEXTU,FIRSTU,VFYDAT.

007036                                  EXALL:: LET R1 := #CMDSEQ               ;INIT SEQUENCE TABLE POINTER.
007036  012701  003554                                                            MOV     #CMDSEQ,R1
007042                                          WHILE (R1) NE #END DO           ;WHILE THERE ARE CMDS IN THE SEQUENCE TABLE.
007042                                                                          50006$:
007042  021127  177777                                                          CMP     (R1),#END
007046  001527                                                                  BEQ     50007$
007050  004737  010000                            JSR   PC,SETUP                ;GO SETUP THE COMMAND BLOCK.
007054                                            WHILE NCNT LT NCNT1 DO        ;WHILE THERE ARE RECORDS REMAINING:
007054                                                                          50010$:
007054  023737  003420  003422                                                  CMP     NCNT,NCNT1
007062  002116                                                                  BGE     50011$
007064  004737  007672                              JSR PC,CMDAC                ;STORE CMD ASCII IN ERROR MESSAGE.
007070                                              IFB RANDOM NE #0 THEN       ;IF IN RANDOM MODE:
007070  105737  003533                                                          TSTB    RANDOM
007074  001435                                                                  BEQ     50012$
007076                                                IF CMDWRD EQ #WRT THEN    ;IF CMD IS A WRITE THEN:
007076  023727  003426  104005                                                  CMP     CMDWRD,#WRT
007104  001031                                                                  BNE     50013$
007106                                                  IFB VFYFLG EQ #0 THEN   ;IF DATA IS NOT TO BE VERIFIED THEN:
007106  105737  003534                                                          TSTB    VFYFLG
007112  001026                                                                  BNE     50014$
007114                                                    LET RANB := RANB + RANS  ;GENERATE
007114  063737  003442  003440                                                  ADD     RANS,RANB
007122                                                    LET RANS := RANS + RANB  ;RANDOM
007122  063737  003440  003442                                                  ADD     RANB,RANS
007130                                                    LET BRFCNT := RANS       ;LENGTH
007130  013737  003442  003424                                                  MOV     RANS,BRFCNT
007136                                                    LET BRFCNT := BRFCNT CLR.BY LENMSK  ;MASK RANDOM LENGTH.
007136  043737  003436  003424                                                  BIC     LENMSK,BRFCNT
007144                                                    IF BRFCNT LT #18. THEN        ;DO NOT ALLOW BYTE COUNT OF LESS THAN 18.
007144  023727  003424  000022                                                  CMP     BRFCNT,#18.
007152  002003                                                                  BGE     50015$
007154                                                      LET BRFCNT := #18.  ;CHANGE COUNT OF 0-17 TO 18.
007154  012737  000022  003424                                                  MOV     #18.,BRFCNT
007162                                                    ENDIF
007162                                                                          50015$:
007162                                                    LET CMDPKT+CP.CNT := BRFCNT  ;MOVE BRF TO CMD PACKET.
007162  013737  003424  002322                                                  MOV     BRFCNT,CMDPKT+CP.CNT
007170                                                  ENDIF
007170
007170                                                ENDIF                     50014$:
007170
007170                                            ENDIF                         50013$:
007170
007170  004737  007332                          JSR PC,EXSUB                  ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATUS.
007174  004737  020274                          JSR PC,CKHAE                  ;CHECK HALT AFTER EACH CMD FLAG.
007200                                        LET R2 := #1                    ;SET ALL UNITS AT BOT/EOT.
007200  012702  000001                                                          MOV     #1,R2
007204  004737  017666                          JSR PC,FIRSTU                 ;FIND FIRST UNIT.
007210                                        WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE UNITS:
```

```
        007210
        007210  026527  002536  177777                                                  50016$:
        007216  001426                                                                    CMP     DEVTBL(R5),#END
   2974 007220                                                                          BEQ     50017$
        007220  032737  000400  003426  IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
        007226  001406                                                                  BIT     #MOD.CO,CMDWRD
   2975 007230                                                                          BEQ     50020$
        007230  032765  000002  003516    IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT THEN:
        007236  001001                                                                  BIT     #XO.BOT,EOTFLG(R5)
   2976 007240                                                                          BNE     50021$
        007240  005002      LET R2 := #0                                                ;CLEAR EOT/BOT FLAG.
   2977 007242                                                                          CLR     R2
        007242    ENDIF
   2978 007242                                                                  50021$:
        007242  000411  ELSE                          ;ELSE IF CMD IS NOT REVERSE:
        007244                                                                          BR      50022$
   2979 007244                                                                  50020$:
        007244  032765  000001  003516    IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
        007252  001404                                                                  BIT     #XO.EOT,EOTFLG(R5)
        007254  032737  000001  003426                                                  BEQ     50023$
        007262  001001                                                                  BIT     #CMD.CO,CMDWRD
        007264                                                                          BNE     50024$
   2980                                                                         50023$:
   2981 007264                                                                ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
        007264  005002      LET R2 := #0                ;CLEAR EOT/BOT FLAG.
   2982 007266                                                                          CLR     R2
        007266    ENDIF
   2983 007266                                                                  50024$:
        007266  ENDIF
   2984 007266  004737  017734                                                  50022$:
   2985 007272    JSR PC,NEXTU                 ;FIND NEXT UNIT
        007272  000746  ENDDO                                                   ;
        007274                                                                          BR      50016$
   2986 007274                                                                  50017$:
        007274  020227  000001  IF R2 EQ #1 THEN          ;IF ALL UNIT ARE AT EOT/BOT THEN:
        007300  001001                                                                  CMP     R2,#1
   2987 007302  000412      BR        EXARTN             ;RETURN WITH R2 = #1.       BNE     50025$
   2988 007304    ENDIF
        007304                                                                  50025$:
   2989 007304                                                                ;UPDATE RECORD COUNT.
        007304  005237  003420      LET NCNT := NCNT + #1                               INC     NCNT
   2990 007310                                                                ;SAVE PREVIOUS COMMAND WORD.
        007310  013737  003426  003432      LET PCMDWD := CMDWRD                        MOV     CMDWRD,PCMDWD
   2991 007316    ENDDO
        007316  000656                                                                  BR      50010$
        007320                                                                  50011$:
   2992 007320  004737  016530      JSR   PC,VFYDAT       ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
   2993                                                    ;VERIFY THE LAST N RECORDS OF DATA.
   2994 007324  ENDDO
        007324  000646                                                                 BR      50006$
        007326                                                                  50007$:
   2995 007326    LET R2 := #0                 ;SET NORMAL RETURN INDICATOR.
        007326  005002                                                                 CLR     R2
   2996 007330  000207  EXARTN: RTS PC         ;RETURN.
   2997
   2998
   2999
```

```
                                ;       SUBROUTINE TO ISSUE COMMAND TO ALL DEVICES, WAIT FOR
                                ;       ALL INTERRUPTS, AND CHECK ALL STATUS.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:          EXCUTE,GOWAIT,NEXTU,FIRSTU.

007332  004737  017666          EXSUB::     JSR PC,FIRSTU                   ;SET UP FOR FIRST UNIT.
007336                                      WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
007336                                                                          50026$:
007336  026527  002536  177777                                                CMP     DEVTBL(R5),#END
007344  001465                                                                  BEQ     50027$
007346                                        IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
007346  032737  000400  003426                                                BIT     #MOD.CO,CMDWRD
007354  001421                                                                  BEQ     50030$
007356                                          IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
007356  032765  000002  003516                                                BIT     #XO.BOT,EOTFLG(R5)
007364  001014                                                                  BNE     50031$
007366                                            IF #XO.EOT SETIN EOTFLG(R5) THEN  ;BUT IF AT EOT
007366  032765  000001  003516                                                BIT     #XO.EOT,EOTFLG(R5)
007374  001406                                                                  BEQ     50032$
007376                                              IFB ALLEOT NE #0 THEN       ;AND ALL OTHERS AT EOT
007376  105737  003543                                                        TSTB    ALLEOT
007402  001402                                                                  BEQ     50033$
007404  004737  010774                                JSR PC,EXCUTE           ;THEN EXECUTE REV CMD
007410                                              ENDIF                     ;IF NOT ALL AT EOT, FREEZE UNIT(S) AT EOT
007410                                                                          50033$:
007410                                            ELSE                        ;IF NOT AT BOT AND
007410  000402                                                                  BR      50034$
007412                                                                          50032$:
007412  004737  010774                             JSR PC,EXCUTE                ;NOT AT EOT, EXEC REV CMD
007416                                            ENDIF
007416
007416                                          ENDIF
007416                                                                          50034$:
007416                                                                          50031$:
007416  000435                                ELSE                        ;ELSE IF CMD IS NOT REVERSE:
007420                                                                          BR      50035$
007420                                                                          50030$:
007420                                          IF CMDLG EQ #2 AND #XO.BOT SETIN EOTFLG(R5) THEN
007420  023727  003434  000002                                                CMP     CMDLG,#2
007426  001011                                                                  BNE     50036$
007430  032765  000002  003516                                                BIT     #XO.BOT,EOTFLG(R5)
007436  001405                                                                  BEQ     50036$
                                                                    ;CLEAR BAD SPOT COUNTS WHEN WRITING FROM BOT
007440                                            LET BTPT := BTADDR(R5)
007440  016537  002550  003526                                                MOV     BTADDR(R5),BTPT
007446                                            LET @BTPT := #0
007446  005077  174054                                                        CLR     @BTPT
007452                                          ENDIF
007452                                                                          50036$:
007452                                          IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
007452  032765  000001  003516                                                BIT     #XO.EOT,EOTFLG(R5)
007460  001404                                                                  BEQ     50037$
007462  032737  000001  003426                                                BIT     #CMD.CO,CMDWRD
007470  001003                                                                  BNE     50040$
007472                                                                          50037$:
                                                                    ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
```

F6


```
3027 007472  004737  010774                  JSR PC,EXCUTE           ;ISSUE CMD TO TK25.
3028 007476                                ELSE
     007476  000405                                                          BR      50041$
     007500                                                                  50040$:
3029 007500                                  IFB ALLEOT NE #0 THEN
     007500  105737  003543                                                  TSTB    ALLEOT
     007504  001402                                                          BEQ     50042$
3030 007506  004737  010774                    JSR PC,EXCUTE
3031 007512                                  ENDIF
     007512                                                                  50042$:
3032 007512                                ENDIF
     007512                                                                  50041$:
3033 007512                              ENDIF
     007512                                                                  50035$:
3034 007512  004737  017734              JSR         PC,NEXTU        ;FIND NEXT UNIT IN TEST CYCLE.
3035 007516                             ENDDO
     007516  000707                                                          BR      50026$
     007520                                                                  50027$:
3036 007520                            IFB RPTFLG NE #0 THEN        ;IF REPORT HAS BEEN REQUESTED THEN:
     007520  105737  003535                                                  TSTB    RPTFLG
     007524  001403                                                          BEQ     50043$
3037 007526                              LET RPTFLG :B= #0            ;CLR THE FLAG.
     007526  105037  003535                                                  CLRB    RPTFLG
3038 007532                              DORPT                        ;PRINT THE PERFORMANCE REPORT.
     007532  104424                                                              TRAP    C$DRPT
3039 007534                            ENDIF                          ;
     007534                                                                  50043$:
3040 007534  004737  017666            JSR PC,FIRSTU                  ;SET UP FOR FIRST UNIT.
3041 007540                            WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
     007540                                                                  50044$:
     007540  026527  002536  177777                                          CMP     DEVTBL(R5),#END
     007546  001450                                                          BEQ     50045$
3042 007550                              IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
     007550  032737  000400  003426                                          BIT     #MOD.CO,CMDWRD
     007556  001421                                                          BEQ     50046$
3043 007560                                IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
     007560  032765  000002  003516                                          BIT     #XO.BOT,EOTFLG(R5)
     007566  001014                                                          BNE     50047$
3044 007570                                  IF #XO.EOT SETIN EOTFLG(R5) THEN    ;BUT IF AT EOT
     007570  032765  000001  003516                                          BIT     #XO.EOT,EOTFLG(R5)
     007576  001406                                                          BEQ     50050$
3045 007600                                    IFB ALLEOT NE #0 THEN            ;AND ALL OTHERS AT EOT
     007600  105737  003543                                                  TSTB    ALLEOT
     007604  001402                                                          BEQ     50051$
3046 007606  004737  011470                      JSR PC,GOWAIT                  ;THEN WAIT FOR CMD END
3047 007612                                    ENDIF                          ;IF NOT ALL AT EOT, DO NOT WAIT
     007612                                                                  50051$:
3048 007612                                  ELSE                             ;NOT AT BOT, AND NOT AT EOT
     007612  000402                                                          BR      50052$
     007614                                                                  50050$:
3049 007614  004737  011470                    JSR PC,GOWAIT                          ;WAIT FOR INT,CHECK STATUS.
3050 007620                                  ENDIF
     007620                                                                  50052$:
3051 007620                                ENDIF
     007620                                                                  50047$:
3052 007620                              ELSE                         ;ELSE IF CMD IS FORWARD:
     007620  000420                                                          BR      50053$
```

```
      007622                                                                50046$:
3053  007622                                  IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
      007622  032765  000001  003516                                        BIT     #XO.EOT,EOTFLG(R5)
      007630  001404                                                        BEQ     50054$
      007632  032737  000001  003426                                        BIT     #CMD.CO,CMDWRD
      007640  001003                                                        BNE     50055$
      007642                                                                50054$:
3054                                                         ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
3055  007642  004737  011470                   JSR PC,GOWAIT  ;WAIT FOR INT,CHECK STATUS.
3056  007646                                  ELSE
      007646  000405                                                        BR      50056$
      007650                                                                50055$:
3057  007650                                   IFB ALLEOT NE #0 THEN
      007650  105737  003543                                                TSTB    ALLEOT
      007654  001402                                                        BEQ     50057$
3058  007656  004737  011470                     JSR PC,GOWAIT
3059  007662                                   ENDIF
      007662                                                                50057$:
3060  007662                                  ENDIF
      007662                                                                50056$:
3061  007662                                 ENDIF
      007662                                                                50053$:
3062  007662  004737  017734                 JSR       PC,NEXTU     ;FIND NEXT UNIT IN TEST CYCLE.
3063  007666                                ENDDO
      007666  000724                                                        BR      50044$
      007670                                                                50045$:
3064  007670  000207                        RTS PC                 ;RETURN.
```

```
                                ;       THIS SUBROUTINE STORES THE ASCII FOR THE CURRENT COMMAND AND PREVIOUS
                                ;       COMMAND IN THE STANDARD ERROR MESSAGE.  ON ENTRY LOCATION CMDWRD
                                ;       CONTAINS CURRENT CMD AND LOCATION PCMDWD CONTAINS PREVIOUS CMD.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R3, R4.
                                ;       CALLS:          GCMDA

007672                          CMDAC:: LET R4 := CMDWRD                ;R4 = CMD BINARY.
007672  013704  003426                                                          MOV     CMDWRD,R4
007676  004737  007746                  JSR     PC,GCMDA                ;GET CMD ASCII.
007702  112337  006154                  MOVB    (R3)+,STAER1+2          ;MOVE CMD ASCII
007706  112337  006155                  MOVB    (R3)+,STAER1+3          ;
007712  111337  006156                  MOVB    (R3),STAER1+4           ;INTO MSG.
007716                                  LET R4 := PCMDWD                ;R4 = PREVIOUS CMD BINARY.
007716  013704  003432                                                          MOV     PCMDWD,R4
007722  004737  007746                  JSR     PC,GCMDA                ;GET CMD ASCII.
007726  000240                          NOP
007730                                  LET STAER7+24 :B= (R3)+         ;MOVE CMD ASCII
007730  112337  006270                                                          MOVB    (R3)+,STAER7+24
007734                                  LET STAER7+25 :B= (R3)+         ;
007734  112337  006271                                                          MOVB    (R3)+,STAER7+25
007740                                  LET STAER7+26 :B= (R3)          ;INTO MSG.
007740  111337  006272                                                          MOVB    (R3),STAER7+26
007744  000207                          RTS     PC                      ;RETURN. GO EXECUTE NEXT FUNCTION.


                                ;       SUBROUTINE TO FIND THE ASCII EQUIVILENT OF THE COMMAND IN R4.
                                ;       ADDRESS OF ASCII 1ST WORD IS RETURNED IN R3.
                                ;       INPUTS:         R4 = PRESENT COMMAND WORD.
                                ;       OUTPUTS:        R3 = ADDRESS OF PRESENT COMMAND ASCII.
                                ;       REGISTERS:
                                ;       CALLS:

007746                          GCMDA:: LET R3 := #0                    ;INIT CMD TBL POINTER.
007746  005003                                                                 CLR     R3
007750                                  WHILE CMDTBL(R3) NE R4 DO       ;UNTIL CURRENT CMD IS FOUND:
007750                                                                  50060$:
007750  026304  003656                                                         CMP     CMDTBL(R3),R4
007754  001403                                                                 BEQ     50061$
007756                                    LET R3 := R3 + #2             ;SEARCH CMD TABLE.
007756  062703  000002                                                         ADD     #2,R3
007762                                  ENDDO
007762  000772                                                                 BR      50060$
007764                                                                  50061$:
007764                                  LET R4 := R3
007764  010304                                                                 MOV     R3,R4
007766                                  LET R3 := R3 SHIFT -1           ;POINT TO ASCII FOR THAT COMMAND
007766  006203                                                                 ASR     R3
007770  060403                          ADD     R4,R3
007772  062703  003744                  ADD     #CMDASC,R3
007776  000207                          RTS     PC                      ;RETURN.
```

```
                                ;       THIS SUBROUTINE LOADS THE TK25 COMMAND PACKET FROM ONE
                                ;       ENTRY IN THE SEQUENCE TABLE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R3.
                                ;       CALLS:          GENPAT.

010000                          SETUP:: LET CMDLG := #0                 ;CLR CMD LOGGING CODE(DISABLES LOGGING)
010000  005037  003434                          CLR     CMDLG
010004  012137  002314                  MOV     (R1)+,CMDPKT            ;LOAD THE COMMAND WORD.
010010  011137  002322                  MOV     (R1),CMDPKT+CP.CNT      ;LOAD THE BYTE/RECORD/FILE COUNT.
010014  011137  003424                  MOV     (R1),BRFCNT             ;SAVE BRF FOR THIS COMMAND.
010020  013702  002314                  MOV     CMDPKT,R2               ;GET CMD.
010024  042702  177740                  BIC     #NCMD.C,R2              ;CLR ALL BUT CMD BITS.
010030  010203                          MOV     R2,R3                   ;SAVE IT TWICE.
010032  162703  000010                  SUB     #CMD.C3,R3              ;POSITION COMMAND?
010036  001003                          BNE     2$                      ;BR IF NOT.
010040  011137  002316                  MOV     (R1),CMDPKT+2           ;MOVE BPCR IN 2ND PKT WORD FOR POSITION CMD.
010044  000461                          BR      3$
010046                          2$:     IF CMDPKT EQ #WTM THEN          ;IF CMD IS A WRITE TAPE MARK THEN:
010046  023727  002314  100011                      CMP     CMDPKT,#WTM
010054  001003                          BNE     50062$
010056                                    LET CMDLG := #2                 ;WTM LOGGING CODE IS 2.
010056  012737  000002  003434                      MOV     #2,CMDLG
010064                                  ENDIF
010064                                                  50062$:
010064  010203                          MOV     R2,R3
010066  162703  000001                  SUB     #CMD.C0,R3              ;IS IT A READ?
010072  001017                          BNE     1$                      ;BR IF NOT.
010074  013737  003416  002316          MOV     DATARD,CMDPKT+CP.ADL    ;IF SO, LOAD THE BUFFER ADDR.
010102                                  IF #MOD.C0 SETIN CMDPKT THEN    ;IF CMD IS A READ REV THEN:
010102  032737  000400  002314                      BIT     #MOD.C0,CMDPKT
010110  001404                          BEQ     50063$
010112                                    LET CMDLG := #4                 ;LOGGING CODE IS 4.
010112  012737  000004  003434                      MOV     #4,CMDLG
010120                                  ELSE                            ;ELSE - IF CMD IS A READ FWD:
010120  000403                                  BR      50064$
010122                                                  50063$:
010122                                    LET CMDLG := #6                 ;LOGGING CODE IS 6.
010122  012737  000006  003434                      MOV     #6,CMDLG
010130                                  ENDIF
010130                                                  50064$:
010130  000427                          BR      3$                      ;CONTINUE.
010132  010203                  1$:     MOV     R2,R3                   ;IS IT
010134  162703  000004                  SUB     #CMD.C2,R3              ;A SET CHARACTERISTICS CMD?
010140  001011                          BNE     4$                      ;BR IF NOT.
010142                                  LET CMDPKT+CP.ADL := #SCHBK     ;SET UP ADR LO FOR SET CHAR.
010142  012737  002446  002316                      MOV     #SCHBK,CMDPKT+CP.ADL
010150  012737  000010  002322          MOV     #SCHCNT,CMDPKT+CP.CNT   ;SET BUFFER EXTENT
010156                                  LET SCHBK+6 := (R1)              ;STORE CHARACTERISTIC CODE IN SCH BLOCK.
010156  011137  002454                              MOV     (R1),SCHBK+6
010162  000412                          BR      3$                      ;CONTINUE.
010164  010203                  4$:     MOV     R2,R3                   ;IS IT
010166  162703  000006                  SUB     #CMD.C1!CMD.C2,R3       ;A DIAGNOSTIC (DIA) CMD?
010172  001006                          BNE     3$                      ;BR IF NOT.
010174  012737  000020  002322          MOV     #DIACNT,CMDPKT+CP.CNT   ;LOAD BUFFER EXTENT.
010202  012737  003414  002316          MOV     #DIABLK,CMDPKT+CP.ADL   ;LOAD BUFFER ADR LOW.
```

```
010210  005721                          3$:     TST     (R1)+                   ;POINT TO N (NUMBER OF TIMES TO EXECUTE THIS INSTRUC
010212                                          LET NCNT1 := (R1)+              ;SAVE NUMBER OF OPERATIONS
010212  012137  003422                                                                  MOV     (R1)+,NCNT1
010216                                          LET NCNT := #0                  ;CLEAR OPERATION COUNTER.
010216  005037  003420                                                                  CLR     NCNT
010222  012137  003454                          MOV     (R1)+,PATERN            ;SAVE PATTERN CODE FOR CURRENT CMD.
010226  010203                                  MOV     R2,R3                   ;IS IT
010230  162703  000005                          SUB     #CMD.CO!CMD.C2,R3       ;A WRITE?
010234  001010                                  BNE     5$                      ;BR IF NOT.
010236  013737  003414  002316                  MOV     DATAWT,CMDPKT+CP.ADL    ;LOAD WRITE BUFFER LO ORDER.
010244  004737  010356                          JSR     PC,GENPAT               ;GO GENERATE THE WRITE PATTERN.
010250                                          LET CMDLG := #2                 ;WRITE LOGGING CODE IS 2.
010250  012737  000002  003434                                                          MOV     #2,CMDLG
010256                                  5$:     IF #VFY.C SETIN CMDPKT THEN     ;IF DATA VERIFICATION IS REQUIRED:
010256  032737  000100  002314                                                          BIT     #VFY.C,CMDPKT
010264  001407                                                                          BEQ     50065$
010266                                            LET VFYFLG :B= #1             ;SET VERIFY FLAG.
010266  112737  000001  003534                                                          MOVB    #1,VFYFLG
010274  042737  000100  002314                    BIC   #VFY.C,CMDPKT           ;CLEAR VERIFY BIT(NOT USED BY HARDWARE).
010302                                          ELSE                            ;IF DATA VERIFICATION IS NOT REQUIRED:
010302  000402                                                                          BR      50066$
010304                                                                                  50065$:
010304                                            LET VFYFLG :B= #0             ;CLR VERIFY FLAG.
010304  105037  003534                                                                  CLRB    VFYFLG
010310                                          ENDIF
010310                                                                                  50066$:
010310                                          LET PCMDWD := CMDWRD            ;SAVE PREVIOUS CMD WORD.
010310  013737  003426  003432                                                          MOV     CMDWRD,PCMDWD
010316                                          LET CMDWRD := CMDPKT            ;SAVE PRESENT CMD WORD.
010316  013737  002314  003426                                                          MOV     CMDPKT,CMDWRD
010324                                          IFB SWBFLG NE #0 THEN           ;IF SWAP BYTES IS ENABLED:
010324  105737  003536                                                                  TSTB    SWBFLG
010330  001403                                                                          BEQ     50067$
010332                                            LET CMDPKT := CMDPKT SET.BY #SWB.C  ;SET SWAP BIT IN COMMAND.
010332  052737  010000  002314                                                          BIS     #SWB.C,CMDPKT
010340                                          ENDIF
010340                                                                                  50067$:
010340  042737  004000  002314                  BIC     #BRF.C,CMDPKT           ;CLR BRF BIT (INTERNAL ONLY).
010346                                          LET CMDSAV := CMDPKT            ;SAVE 1ST WORD OF COMMAND PACKET.
010346  013737  002314  003430                                                          MOV     CMDPKT,CMDSAV
010354  000207                                  RTS     PC                      ;RETURN.
```

```
                                ;       THIS SUBROUTINE SETS UP AND CALLS THE APPROPRIATE SUBROUTINE TO GENERATE
                                ;       THE DESIRED PATTERN FOR THE WRITE AND WRITE/VERIFY COMMANDS.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R3, R4.
                                ;       CALLS:          PATR0 - PATR7

010356                          GENPAT:: LET R3 := PATERN SHIFT 1       ;SETUP PATTERN ROUTINE POINTER
010356  013703  003454                                                          MOV     PATERN,R3
010362  006303                                                                  ASL     R3
010364                                  LET R4 := BRFCNT + #1   ;SET LENGTH OF WRITE BFR
010364  013704  003424                                                          MOV     BRFCNT,R4
010370  005204                                                                  INC     R4
010372                                  LET R4 := R4 CLR.BY #1  ;ROUNDED UP TO NEXT WORD
010372  042704  000001                                                          BIC     #1,R4
010376                                  LET R4 := R4 - #2       ;WITH FIRST WORD RESERVED
010376  162704  000002                                                          SUB     #2,R4
010402                                  LET R2 := DATAWT + #2   ;FOR RECORD COUNT
010402  013702  003414                                                          MOV     DATAWT,R2
010406  062702  000002                                                          ADD     #2,R2
010412  004773  010420                  JSR     PC,@PATTBL(R3)  ;GO GENERATE THE APPROPRIATE PATTERN.
010416  000207                          RTS     PC              ;RETURN TO SETUP SUBROUTINE.

                                        ;TK25 WRITE PATTERN LOOKUP TABLE.  USED TO JSR TO THE
                                        ;CORRECT DATA PATTERN GENERATING ROUTINE.

010420  010442                  PATTBL: PATR0                   ; INCREMENTING PATTERN, 0 - 377
010422  010500                          PATR1                   ; ALL ONES PATTERN
010424  010520                          PATR2                   ; ALL ZEROES PATTERN
010426  010530                          PATR3                   ; '1' BIT SHIFT, RIGHT TO LEFT
010430  010554                          PATR4                   ; '0' BIT SHIFT, RIGHT TO LEFT
010432  010566                          PATR5                   ; ALTERNATE '0' & '1' WITH ALT. BYTES COMPL.
010434  010600                          PATR6                   ; ALTERNATE BYTES OF 000 AND 377
010436  010620                          PATR7                   ; RANDOM PATTERN.
010440  010744                          PATR8                   ; DUMMY.  NO PATTERN, JUST EXITS.


                                        ;INCREMENTING PATTERN.  0 - 377.

010442                          PATR0:: LET R3 := # ^O
010442  012703  000400                                                          MOV     #400,R3
010446                          1$:     LET R4 :- R4 - #2       ;DECREMENT WORD COUNT.
010446  162704  000002                                                          SUB     #2,R4
010452  100411                          BMI     2$              ;BR IF DONE.
010454                                  LET (R2)+ := R3         ;STORE DATA WORD.
010454  010322                                                                  MOV     R3,(R2)+
010456                                  LET R3 := R3 + #1002    ;UPDATE PATTERN.
010456  062703  001002                                                          ADD     #1002,R3
010462                                  IF R3 EQ #1000 THEN     ;IF PATTERN HAS WRAPPED AROUND THEN:
010462  020327  001000                                                          CMP     R3,#1000
010466  001002                                                                  BNE     50070$
010470                                    LET R3 := #400          ;INIT THE PATTERN AGAIN.
010470  012703  000400                                                          MOV     #400,R3
010474                                  ENDIF
010474                                                                  50070$:
010474  000764                          BR      1$              ;DO IT AGAIN.
010476  000207                  2$:     RTS     PC              ;RETURN.
```

```
                                        ;ALL ONE'S PATTERN.
                                        
010500  012703  177777          PATR1:: MOV     #-1,R3          ;ALL ONES PATTERN;.
010504                          ZROPAT: LET R4 := R4 - #2       ;DECREMENT BYTE COUNT.
010504  162704  000002                                                  SUB     #2,R4
010510  100402                          BMI     1$              ;DONE?,BR IF YES.
010512  010322                          MOV     R3,(R2)+        ;IF NOT LOAD NEXT BYTE WITH PATTERN.
010514  000773                          BR      ZROPAT          ;DO IT AGAIN.
                                        
010516  000207                  1$:     RTS     PC              ;RETURN.
```

```
3226                                    ;ALL ZEROES PATTERN.
3227
3228 010520  005003                PATR2:: CLR     R3              ;CLR PATTERN REGISTER.
3229 010522  004737  010504                JSR     PC,ZROPAT       ;GO GENERATE IT.
3230 010526  000207                        RTS     PC              ;RETURN.
3231
3232                                    ;ONE BIT WALKING FROM R TO L IN A FIELD OF ZEROES.
3233
3234 010530  012703  000401        PATR3:: MOV     #401,R3         ;INIT PATTERN REGISTER.
3235 010534                        WLKZRO: LET R4 := R4 - #2       ;DECREMENT WORD COUNT.
     010534  162704  000002                                                  SUB     #2,R4
3236 010540  100404                        BMI     1$              ;BR IF DONE.
3237 010542  010322                        MOV     R3,(R2)+        ;LOAD DATA.
3238 010544  006303                        ASL     R3              ;SHIFT PATTERN.
3239 010546  005503                        ADC     R3              ;ADD CARRY BACK INTO PATTERN.
3240 010550  000771                        BR      WLKZRO          ;DO IT AGAIN.
3241 010552  000207                1$:     RTS     PC              ;RETURN.
3242
3243                                    ;ZERO BIT WALKING FROM R TO L IN A FIELD OF 1'S.
3244
3245 010554  012703  177376        PATR4:: MOV     #177376,R3      ;INIT PATTERN REGISTER.
3246 010560  004737  010534                JSR     PC,WLKZRO       ;GO GENERATE ;IT.
3247 010564  000207                        RTS     PC              ;RETURN.
3248
3249                                    ;ALTERNATING ONE AND ZERO BITS WITH ALTERNATE BYTES
3250                                    ;COMPLEMENTED.
3251
3252 010566  012703  125125        PATR5:: MOV     #125125,R3      ;INIT PATTERN REGISTER.
3253 010572  004737  010504                JSR     PC,ZROPAT       ;GO GENERATE IT.
3254 010576  000207                        RTS     PC              ;RETURN.
3255
3256                                    ;ALTERNATING BYTES OF 000 AND 377.
3257
3258 010600  012703  177400        PATR6:: MOV     #177400,R3      ;INIT PATTERN REGISTER.
3259 010604                        1$:     LET R4 := R4 - #2       ;DECREMENT WORD COUNT.
     010604  162704  000002                                                  SUB     #2,R4
3260 010610  100402                        BMI     2$              ;BR IF DONE.
3261 010612  010322                        MOV     R3,(R2)+        ;LOAD DATA.
3262 010614  000773                        BR      1$              ;DO IT AGAIN.
3263 010616  000207                2$:     RTS     PC              ;RETURN.
3264
3265                                    ;RANDOM PATTERN GENERATOR
3266
3267 010620                        PATR7::
3268 010620  012737  001233  010736        MOV     #RS1,RAN1       ;SET UP THE SEED
3269 010626  012737  007622  010740        MOV     #RS2,RAN2       ;SET UP THE SEED
3270 010634  012737  000000  010742        MOV     #RS3,RAN3       ;SET UP THE SEED
3271 010642                        1$:     LET R4 := R4 - #2       ;DECREMENT WORD COUNT
     010642  162704  000002                                                  SUB     #2,R4
3272 010646  100432                        BMI     GIT             ;BR IF DONE.
3273 010650  010346                        MOV     R3,-(SP)
3274 010652  013703  010736                MOV     RAN1,R3         ;MOVE THE FIRST SEED INTO R3
3275 010656  000241                        CLC                     ;CLEAR THE CARRY FLAG
3276 010660  005337  010742                DEC     RAN3            ;DECREMENT THE THIRD SEED
3277 010664  006103                        ROL     R3              ;
3278 010666  006103                        ROL     R3              ;
3279 010670  063703  010740                ADD     RAN2,R3         ;ADD THE SECOND SEED TO R3
```

```
 3280 010674  010337  010736                MOV     R3,RAN1         ;PUT IT ALL IN THE FIRST SEED
 3281 010700  063703  010742                ADD     RAN3,R3         ;PUT THE THIRD SEED INTO R3
 3282 010704  006103                        ROL     R3              ;
 3283 010706  006103                        ROL     R3              ;
 3284 010710  063703  010740                ADD     RAN2,R3         ;ADD THE SECOND SEED TO R3
 3285 010714  006103                        ROL     R3              ;
 3286 010716  006103                        ROL     R3              ;
 3287 010720  010337  010740                MOV     R3,RAN2         ;PUT IT IN THE SECONG SEED
 3288 010724  012603                        MOV     (SP)+,R3        ;RESTORE R3
 3289 010726  013722  010740                MOV     RAN2,(R2)+      ;PUT # IN BUFFER
 3290 010732  000743                        BR      1$              ;CONTINUE
 3291 010734  000207              GIT:      RTS     PC              ;RETURN
 3292
 3293 010736  000000              RAN1::    .WORD   0
 3294 010740  000000              RAN2::    .WORD   0
 3295 010742  000000              RAN3::    .WORD   0
 3296         001233              RS1       ==      1233
 3297         007622              RS2       ==      7622
 3298         000000              RS3       ==      0
 3299
 3300                             ;         NO PATTERN GENERATION.
 3301
 3302 010744  000207              PATR8::   RTS     PC              ;RETURN.
 3303
 3304 010746                      BGNSRV    CLKSVC
      010746                      CLKSVC::
 3305 010746                              LET TIME1 := TIME1 - #1
      010746  005337  003444                                                  DEC     TIME1
 3306 010752                              LET @LCSR := #0
      010752  005077  172506                                                  CLR     @LCSR
 3307 010756                      ENDSRV    #0
      010756                      L10010:
      010756  142766  000340  000002                                         BICB    #340,2(SP)
      010764  152766  000000  000002                                         BISB    #0,2(SP)
      010772  000002                                                         RTI
```

```
                        ;       THIS SUBROUTINE INITIATES TK25 COMMAND EXECUTION
                        ;       AND CHECKS FOR TK25 RESPONSE.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2, R3.
                        ;       CALLS:          DROPU, MOVMSG, FIRSTU, NEXTU, WSSR.

                        EXCUTE::IF LCSR EQ #0 THEN
010774  005737  003464                                          TST     LCSR
011000  001004                                                  BNE     50071$
                         LET TIME1 := #-1
011002  012737  177777  003444                                  MOV     #-1,TIME1
                         ELSE
011010  000403                                                  BR      50072$
                                                                50071$:
                         LET TIME1 := #304
011012  012737  000304  003444                                  MOV     #304,TIME1
                        ENDIF
                                                                50072$:
                        REPEAT                                  ;WAIT -
                                                                50073$:
                          IF LCSR EQ #0 THEN
011020  005737  003464                                          TST     LCSR
011024  001003                                                  BNE     50074$
                           LET TIME1 := TIME1 - #1
011026  005337  003444                                          DEC     TIME1
                          ELSE
011032  000403                                                  BR      50075$
                                                                50074$:
                           LET @LCSR := #100                    ;ARM THE INTERRUPTS
011034  012777  000100  172422                                  MOV     #100,@LCSR
                           ENDIF
011042
                                                                50075$:
                          IF TIME1 EQ #0 THEN                   ;IF TIMED OUT:
011042  005737  003444                                          TST     TIME1
011046  001011                                                  BNE     50076$
                            JSR PC,MOVMSG                       ;MOVE CURRENT PACKET MSG.
                            ERRDF 2,NSSRM,STAERM                ;REPORT TK25 NOT READY
011054  104455                                                  TRAP    C$ERDF
011056  000002                                                  .WORD   2
011060  004365                                                  .WORD   NSSRM
011062  005640                                                  .WORD   STAERM
                            JSR PC,DROPU                        ;DROP THE UNIT.
                            BR  EXCRTN                          ;RETURN.
                          ENDIF
                                                                50076$:
                          BREAK
011072  104422                                                  TRAP    C$BRK
                        UNTIL #TS.SSR SETIN @TSSR(R5)           ;WAIT UNTIL DEVICE IS READY.
011074  032775  000200  002466                                  BIT     #TS.SSR,@TSSR(R5)
011102  001746                                                  BEQ     50073$

                        IF CMDWRD EQ #SCH THEN                  ;IF WE ARE DOING A SET CHAR CMD THEN:
011104  023727  003426  140004                                  CMP     CMDWRD,#SCH
011112  001022                                                  BNE     50077$
                          LET R5SAVE := R5                      ;SAVE CURRENT DEVICE POINTER.
011114  010537  003460                                          MOV     R5,R5SAVE
                          JSR   PC,FIRSTU                       ;FIND FIRST UNIT.
```

C7


```
011124                                     WHILE DEVTBL(R5) NE #END DO
011124                                                                              50100$:
011124  026527  002536  177777                                                      CMP     DEVTBL(R5),#END
011132  001405                                                                      BEQ     50101$
011134  004737  012124                        JSR PC,WSSR                     ;WAIT FOR UNIT READY OR TIME OUT,
011140  004737  017734                        JSR PC,NEXTU                    ;FIND NEXT UNIT.
011144                                     ENDDO
011144  000767                                                                      BR      50100$
011146                                                                              50101$:
011146                                     LET R5 := R5SAVE                ;RESTORE CURRENT DEVICE POINTER.
011146  013705  003460                                                              MOV     R5SAVE,R5
011152                                     LET SCHBK := MSGPKA(R5)         ;SET UP ADR OF MSG PKT IN SCH BLOCK.
011152  016537  002506  002446                                                      MOV     MSGPKA(R5),SCHBK
011160                                  ENDIF
011160                                                                              50077$:
011160                                  LET R3 := MSGPKA(R5)               ;ADR OF THIS UNIT'S MSG PACKET.
011160  016503  002506                                                              MOV     MSGPKA(R5),R3
011164                                  LET R2 := #0                       ;CLR COUNTER.
011164  005002                                                                      CLR     R2
011166                                  WHILE R2 NE #MSGCNT DO             ;WHILE THERE ARE MORE LOCATIONS:
011166                                                                              50102$:
011166  020227  000016                                                              CMP     R2,#MSGCNT
011172  001405                                                                      BEQ     50103$
011174                                    LET (R3)+ := #-1                 ;INIT THE MSG PACKET WITH ALL 1'S
011174  012723  177777                                                              MOV     #-1,(R3)+
011200                                    LET R2 := R2 + #2                ;UPDATE COUNTER.
011200  062702  000002                                                              ADD     #2,R2
011204                                  ENDDO
011204  000770                                                                      BR      50102$
011206                                                                              50103$:
011206  105737  002212                  TSTB    DINT                       ;ARE INTERRUPTS DISABLED.
011212  001023                          BNE     1$                         ;BR IF YES.
011214                                  IFB INTFLG(R5) GT #1 THEN          ;IF MORE THAN ONE INTERRUPT HAS OCCURED:
011214  126527  003506  000001                                                      CMPB    INTFLG(R5),#1
011222  003412                                                                      BLE     50104$
011224                                    LET TSSREG := @TSSR(R5)                  ;FREEZE THE CURRENT STATUS REG FOR PRINT
011224  017537  002466  003462                                                      MOV     @TSSR(R5),TSSREG
011232                                    ERRDF 15,TOOMM,STAERM    ;REPORT TOO MANY INTERRUPTS.
011232  104455                                                                              TRAP    C$ERDF
011234  000017                                                                              .WORD   15
011236  004554                                                                              .WORD   TOOMM
011240  005640                                                                              .WORD   STAERM
011242  004737  017770                    JSR PC,DROPU                       ;DROP THE UNIT
011246  000507                            BR  EXCRTN                         ;RETURN - UNIT HAS BEEN DROPPED.
011250                                  ENDIF
011250                                                                              50104$:
011250                                  LET INTFLG(R5) := #0               ;CLR INTERRUPT FLAG FOR THIS DEV.
011250  005065  003506                                                              CLR     INTFLG(R5)
011254  052737  000200  002314          BIS     #IE.C,CMDPKT               ;SET INT ENABLE BIT.
011262                          1$:     IFB ERRREC EQ #0 THEN              ;IF NOT RETRYING
011262  105737  003505                                                              TSTB    ERRREC
011266  001005                                                                      BNE     50105$
011270                                    LET RECCNT(R5) := RECCNT(R5) + #1
011270  005265  003330                                                              INC     RECCNT(R5)
011274                                    LET @DATAWT := RECCNT(R5)        ;THEN UPDATE REC COUNT TO WRITE IT ON TAPE
011274  016577  003330  172112                                                      MOV     RECCNT(R5),@DATAWT
011302                                  ENDIF
```

D7


```
 011302                                                              50105$:
 011302                               IF L$TEST EQ #3 AND CMDWRD EQ #WRT THEN
 011302  023727  002114  000003                                      CMP     L$TEST,#3
 011310  001047                                                      BNE     50106$
 011312  023727  003426  104005                                      CMP     CMDWRD,#WRT
 011320  001043                                                      BNE     50106$
 011322                               LET LOOPCT :B= LOOPCT + #1
 011322  105237  003537                                              INCB    LOOPCT
 011326                                IFB LOOPCT GT 3 THEN
 011326  123737  003537  000003                                      CMPB    LOOPCT,3
 011334  003435                                                      BLE     50107$
 011336                                    LET LOOPCT :B= #0
 011336  105037  003537                                              CLRB    LOOPCT
 011342                                    LET TIME1 := #4
 011342  012737  000004  003444                                      MOV     #4,TIME1
 011350                                    REPEAT
 011350                                                              50110$:
 011350                                    IF LCSR NE #0 THEN
 011350  005737  003464                                              TST     LCSR
 011354  001404                                                      BEQ     50111$
 011356                                     LET @LCSR := #100
 011356  012777  000100  172100                                      MOV     #100,@LCSR
 011364                                    ELSE
 011364  000416                                                      BR      50112$
 011366                                                              50111$:
 011366                                     DELAY 21.
 011366  012727  000025                                                      MOV     #21.,(PC)+
 011372  000000                                                              .WORD   0
 011374  013727  002116                                                      MOV     L$DLY,(PC)+
 011400  000000                                                              .WORD   0
 011402  005367  177772                                                      DEC     -6(PC)
 011406  001375                                                              BNE     .-4
 011410  005367  177756                                                      DEC     -22(PC)
 011414  001367                                                              BNE     .-20
 011416                                     LET TIME1 := #0
 011416  005037  003444                                              CLR     TIME1
 011422                                    ENDIF
 011422                                                              50112$:
 011422                                    UNTIL TIME1 EQ #0
 011422  005737  003444                                              TST     TIME1
 011426  001350                                                      BNE     50110$
 011430                                ENDIF
 011430                                                              50107$:
 011430                               ENDIF
 011430                                                              50106$:
 011430  012775  002314  002456       MOV     #CMDPKT,@TSDB(R5)      ;LOAD TSDB WITH CMDPKT ADDRESS
                                                                     ;THIS INITIATES COMMAND EXECUTION.
 011436                               IF #TS.SSR SETIN @TSSR(R5) THEN ;IF READY DID NOT DROP THEN:
 011436  032775  ^00200  002466                                      BIT     #TS.SSR,@TSSR(R5)
 011444  001410                                                      BEQ     50113$
 011446  004737  . ?24                  JSR   PC,MOVMSG               ;MOVE CURRENT MESSAGE PACKET TO COMMON.
 011452                                 ERRDF 3,TOERM,STAERM          ;REPORT NO TK25 RESPONSE.
 011452  104455                                                              TRAP    C$ERDF
 011454  000003                                                              .WORD   3
 011456  004303                                                              .WORD   TOERM
 011460  005640                                                              .WORD   STAERM
 011462  004737  017770                 JSR PC,DROPU                  ;DROP THE UNIT
```

```
3386 011466                             ENDIF
     011466                                                                  50113$:
3387 011466  000207                EXCRTN: RTS     PC                       ;RETURN.
3388
3389                                ;       THIS SUBROUTINE WAITS FOR THE TK25 INERRUPT OR DONE BIT TO SET AND ALLOWS THE
3390                                ;       OPERATOR TO TRANSFER CONROL TO THE SUPERVISOR.
3391                                ;       UPON APPEARANCE OF THE INTERRUPT OR DONE, CHECK TSSR FOR STATUS ERRORS,
3392                                ;       LOG BYTES AND ERRORS AND PERFORM ERROR RECOVERY IF NESSASARY.
3393                                ;       INPUTS:
3394                                ;       OUTPUTS:
3395                                ;       REGISTERS:      R2, R3.
3396                                ;       CALLS:          DROPU, MOVMSG, RECUD, CHKERR, LOG, CLRERR.
3397
3398 011470                        GOWAIT::IF DEVTBL(R5) EQ #NINUSE THEN
     011470  026527  002536  177774                                          CMP     DEVTBL(R5),#NINUSE
     011476  001002                                                          BNE     50114$
3399 011500  000137  012106                JMP     1$
3400 011504                             ENDIF
     011504                                                                  50114$:
3401 011504                             IF LCSR EQ #0 THEN
     011504  005737  003464                                                  TST     LCSR
     011510  001004                                                          BNE     50115$
3402 011512                              LET TIME1 := #-1                   ;INIT TIME OUT COUNTER.
     011512  012737  177777  003444                                          MOV     #-1,TIME1
3403 011520                             ELSE
     011520  000403                                                          BR      50116$
     011522                                                                  50115$:
3404 011522                              LET TIME1 := #37777                ;LINE CLOCK STALL VALUE
     011522  012737  037777  003444                                          MOV     #37777,TIME1
3405 011530                             ENDIF
     011530                                                                  50116$:
3406 011530                             REPEAT                             ;REPEAT UNTIL INTERRUPT OCCURES:
     011530                                                                  50117$:
3407 011530                              BREAK                              ;HONOR SUPERVISOR BREAKS
     011530  104422                                                          TRAP    C$BRK
3408 011532                              IF PCMDWD EQ #RWD AND CMDWRD EQ #WRT THEN
     011532  023727  003432  102010                                          CMP     PCMDWD,#RWD
     011540  001020                                                          BNE     50120$
     011542  023727  003426  104005                                          CMP     CMDWRD,#WRT
     011550  001014                                                          BNE     50120$
3409                                                                        ;POSSIBLE FIRST WRITE ON TAPE
3410 011552                              DELAY 40.
     011552  012727  000050                                                  MOV     #40.,(PC)+
     011556  000000                                                          .WORD   0
     011560  013727  002116                                                  MOV     L$DLY,(PC)+
     011564  000000                                                          .WORD   0
     011566  005367  177772                                                  DEC     -6(PC)
     011572  001375                                                          BNE     .-4
     011574  005367  177756                                                  DEC     -22(PC)
     011600  001367                                                          BNE     .-20
3411 011602                             ENDIF                              ;SO DELAY TO CALIBRATE TAPE
     011602                                                                  50120$:
3412 011602                              IF CMDWRD EQ #RWD THEN             ;IF COMMAND WAS REWIND THEN:
     011602  023727  003426  102010                                          CMP     CMDWRD,#RWD
     011610  001014                                                          BNE     50121$
3413 011612                                DELAY 10.                        ;WAIT  EXTRA 10 MSECS EACH LOOP.
     011612  012727  000012                                                  MOV     #10.,(PC)+
```

F7


```
     011616  000000                                                       .WORD   0
     011620  013727  002116                                               MOV     L$DLY,(PC)+
     011624  000000                                                       .WORD   0
     011626  005367  177772                                               DEC     -6(PC)
     011632  001375                                                       BNE     .-4
     011634  005367  177756                                               DEC     -22(PC)
     011640  001367                                                       BNE     .-20
3414 011642                          ENDIF
     011642                                                       50121$:
3415 011642                          IF CMDWRD EQ #SFF OR CMDWRD EQ #SFR THEN
     011642  023727  003426  105010                               CMP     CMDWRD,#SFF
     011650  001404                                               BEQ     50122$
     011652  023727  003426  105410                               CMP     CMDWRD,#SFR
     011660  001014                                               BNE     50123$
     011662                                                       50122$:
3416 011662                            DELAY 12.               ;ADD DELAY FOR SPACE TAPE MARK COMMANDS
     011662  012727  000014                                               MOV     #12.,(PC)+
     011666  000000                                                       .WORD   0
     011670  013727  002116                                               MOV     L$DLY,(PC)+
     011674  000000                                                       .WORD   0
     011676  005367  177772                                               DEC     -6(PC)
     011702  001375                                                       BNE     .-4
     011704  005367  177756                                               DEC     -22(PC)
     011710  001367                                                       BNE     .-20
3417 011712                          ENDIF
     011712                                                       50123$:
3418 011712                          IFB DINT EQ #0 THEN        ;IF INTERRUPTS ARE ENABLED.
     011712  105737  002212                                       TSTB    DINT
     011716  001003                                               BNE     50124$
3419 011720                            LET R2 := INTFLG(R5)     ;FETCH INTERRUPT OCCURRED FLAG.
     011720  016502  003506                                       MOV     INTFLG(R5),R2
3420 011724                          ELSE                       ;IF IN BRUTUS MODE:
     011724  000406                                               BR      50125$
     011726                                                       50124$:
3421 011726                            LET R3 := COMP #TS.SSR   ;SET UP A MASK FOR THE DONE BIT.
     011726  012703  000200                                       MOV     #TS.SSR,R3
     011732  005103                                               COM     R3
3422 011734                            LET R2 := @TSSR(R5) CLR.BY R3  ;FETCH DONE BIT.
     011734  017502  002466                                       MOV     @TSSR(R5),R2
     011740  040302                                               BIC     R3,R2
3423 011742                          ENDIF
     011742                                                       50125$:
3424 011742                          IF LCSR EQ #0 THEN
     011742  005737  003464                                       TST     LCSR
     011746  001003                                               BNE     50126$
3425 011750                           LET TIME1 := TIME1 - #1   ;UPDATE TIMEOUT COUNTER.
     011750  005337  003444                                       DEC     TIME1
3426 011754                          ELSE
     011754  000403                                               BR      50127$
     011756                                                       50126$:
3427 011756                           LET @LCSR := #100         ;ARM THE INTERRUPTS
     011756  012777  000100  171500                               MOV     #100,@LCSR
3428 011764                          ENDIF
     011764                                                       50127$:
3429 011764                        UNTIL R2 NE #0 OR TIME1 EQ #0   ;REPEAT UNTIL INTERRUPT OR READY OCCURES.
     011764  005702                                               TST     R2
     011766  001003                                               BNE     50130$
```

```
        011770  005737  003444                                                  TST     TIME1
        011774  001255                                                          BNE   . 50117$
        011776                                                                  50130$:
  3430  011776                        IF TIME1 EQ #0 THEN             ;IF TIME OUT HAS OCCURRED:
        011776  005737  003444                                                  TST     TIME1
        012002  001022                                                          BNE     50131$
  3431  012004                          LET @DATAWT := RECCNT(R5) - #1 ;RE-ADJUST REC COUNT DOWN
        012004  016577  003330  171402                                          MOV     RECCNT(R5),@DATAWT
        012012  005377  171376                                                  DEC     @DATAWT
  3432  012016  004737  012224          JSR   PC,MOVMSG                 ;MOVE CURRENT MSG PACKET TO COMMON AREA.
  3433  012022                          ERRDF 4,NOINTM,STAERM ;REPORT NO INTERRUPT.
        012022  104455                                                          TRAP    C$ERDF
        012024  000004                                                          .WORD   4
        012026  004515                                                          .WORD   NOINTM
        012030  005640                                                          .WORD   STAERM
  3434  012032  004737  017770          JSR PC,DROPU            ;DROP THE UNIT.
  3435  012036                          LET R3 := #ENDERF
        012036  012703  003506                                                  MOV     #ENDERF,R3
  3436  012042  004737  012110          JSR PC,CLRERR                   ;CLEAR ALL ERROR FLAGS
  3437  012046                        ELSE
        012046  000417                                                          BR      50132$
        012050                                                                  50131$:
  3438  012050  004737  012224          JSR   PC,MOVMSG                 ;MOVE CURRENT MSG. PACKET TO COMMON AREA.
  3439  012054  004737  012310          JSR   PC,RECUD                  ;UPDATE THE RECORD COUNT.
  3440  012060  004737  012456          JSR   PC,CHKERR                 ;CHECK FOR STATUS ERRORS.
  3441  012064                          IFB WRTYFG EQ #0 THEN           ;
        012064  105737  003477                                                  TSTB    WRTYFG
        012070  001006                                                          BNE     50133$
  3442  012072  004737  016024            JSR PC,LOG                  ;LOG BYTES AND ERRORS.
  3443  012076                            LET R3 := #ENDERF
        012076  012703  003506                                                  MOV     #ENDERF,R3
  3444  012102  004737  012110            JSR PC,CLRERR                       ;CLEAR ALL ERROR FLAGS
  3445  012106                          ENDIF
        012106                                                                  50133$:
  3446  012106                        ENDIF
        012106                                                                  50132$:
  3447  012106  000207          1$:    RTS     PC                      ;RETURN IF DONE.
```

```
                                ;       SUBROUTINE TO CLEAR FLAGS.
                                ;       INPUTS:         R3 = LWA TO BE CLEARED + 2.
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2
                                ;       CALLS:

012110                          CLRERR:: LET R2 := #BGNFLG
012110  012702  003474                                                  MOV     #BGNFLG,R2
                                         REPEAT
012114                                                                  50134$:
                                           LET (R2)+ := #0
012114  005022                                                          CLR     (R2)+
                                         UNTIL R2 EQ R3
012116  020203                                                          CMP     R2,R3
012120  001375                                                          BNE     50134$
012122  000207                         RTS PC


                                ;       SUBROUTINE TO WAIT UNTIL CURRENT UNIT IS READY OR UNTIL TIME OUT.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

012124                          WSSR::  LET TIME1 := #-1                ;INIT TIMEOUT COUNTER.
012124  012737  177777  003444                                                  MOV     #-1,TIME1
012132                                  REPEAT                          ;REPEAT UNTIL DEV READY OR TIMEOUT:
012132                                                                          50135$:
012132  104422                            BREAK                           ;BREAK TO THE SUPERVISOR.
                                                                                TRAP    C$BRK
012134                                   IF LCSR EQ #0 THEN
012134  005737  003464                                                          TST     LCSR
012140  001003                                                                  BNE     50136$
                                          LET TIME1 := TIME1 - #1        ;UPDATE TIMEOUT COUNTER.
012142  005337  003444                                                          DEC     TIME1
                                        ELSE
012146  000403                                                                  BR      50137$
012150                                                                          50136$:
                                          LET @LCSR := #100              ;ARM CLOCK INTERRUPTS
012150  012777  000100  171306                                                  MOV     #100,@LCSR
                                        ENDIF
012156                                                                          50137$:
                                UNTIL #TS.SSR SETIN @TSSR(R5) OR TIME1 EQ #0
012156  032775  000200  002466                                                  BIT     #TS.SSR,@TSSR(R5)
012164  001003                                                                  BNE     50140$
012166  005737  003444                                                          TST     TIME1
012172  001357                                                                  BNE     50135$
012174                                                                          50140$:
                                                                        ;REPEAT UNTIL DEV READY OR TIMEOUT.
012174                          IF TIME1 EQ #0 THEN                     ;IF SYSTEM TIMED OUT THEN
012174  005737  003444                                                          TST     TIME1
012200  001010                                                                  BNE     50141$
012202  004737  012224          JSR PC, MOVMSG                          ;MOVE CURRENT PACKET MSG
012206                          ERRDF   20,NSSRM,STAERM                 ;SSR DIDN'T SET IN TIME
012206  104455                                                                  TRAP    C$ERDF
012210  000024                                                                  .WORD   20
```

```
        012212  004365                                                          .WORD   NSSRM
        012214  005640                                                          .WORD   STAERM
 3482   012216  004737  017770          JSR PC, DROPU                   ;THROW THE UNIT ON THE FLOOR
 3483   012222                          ENDIF
        012222                                                                  50141$:
 3484   012222  000207                  RTS PC                          ;RETURN.
 3485
 3486
 3487                           ;       SUBROUTINE TO MOVE THE CURRENT MESSAGE PACKET TO THE COMMON AREA AND
 3488                           ;       TO UPDATE THE CURRENT TERMINATION CLASS CODE.
 3489                           ;       INPUTS:
 3490                           ;       OUTPUTS:
 3491                           ;       REGISTERS:      R2, R3.
 3492                           ;       CALLS:
 3493
 3494   012224                  MOVMSG:: LET TSSREG := @TSSR(R5)        ;FREEZE THE STATUS REG CONTENTS
        012224  017537  002466  003462                                          MOV     @TSSR(R5),TSSREG
 3495   012232                          LET R2 := TSSREG CLR.BY #TSC.TCC ;EXTRACT THE TERMINATION CLASS CODE.
        012232  013702  003462                                                  MOV     TSSREG,R2
        012236  042702  177761                                                  BIC     #TSC.TCC,R2
 3496   012242                          LET CTCC := R2 SHIFT -1         ;AND SAVE IT
        012242  010237  003456                                                  MOV     R2,CTCC
        012246  006237  003456                                                  ASR     CTCC
 3497   012252                          LET R3 := MSGPKA(R5)            ;ADR OF THIS DEVICE'S MSG.
        012252  016503  002506                                                  MOV     MSGPKA(R5),R3
 3498   012256                          LET R2 := #0                    ;CLR COUNTER.
        012256  005002                                                          CLR     R2
 3499   012260                          WHILE R2 NE #MSGCNT DO          ;WHILE THERE ARE MORE LOCATIONS:
        012260                                                                  50142$:
        012260  020227  000016                                                  CMP     R2,#MSGCNT
        012264  001405                                                          BEQ     50143$
 3500   012266                            LET MSGPKT(R2) := (R3)+       ;MOVE MSG TO COMMON AREA.
        012266  012362  002340                                                  MOV     (R3)+,MSGPKT(R2)
 3501   012272                            LET R2 := R2 + #2             ;UPDATE COUNTER.
        012272  062702  000002                                                  ADD     #2,R2
 3502   012276                          ENDDO
        012276  000770                                                          BR      50142$
        012300                                                                  50143$:
 3503   012300                          LET EOTFLG(R5) := MSGPKT+MS.XSO ;MOVE XSTATO TO EOT FLAG.
        012300  013765  002346  003516                                          MOV     MSGPKT+MS.XSO,EOTFLG(R5)
 3504   012306  000207                  RTS     PC
```

```
                                ;       SUBROUTINE TO ADJUST THE RECORD COUNT.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

                                RECUD:: IFB RECLOG EQ #0 THEN           ;IF RECORD HAS NOT BEEN LOGGED:
012310  105737  003501                                                        TSTB    RECLOG
012314  001057                                                            BNE     50144$
                                  LET RECCNT(R5) := RECCNT(R5) - #1
012316  005365  003330                                                    DEC     RECCNT(R5)
                                  IF #BIT0 NOTSETIN CTCC AND #X2.OPM SETIN MSGPKT+MS.XS2 THEN   ;IF TAPE MOVED THEN:
012322  032737  000001  003456                                            BIT     #BIT0,CTCC
012330  001046                                                            BNE     50145$
012332  032737  100000  002352                                            BIT     #X2.OPM,MSGPKT+MS.XS2
012340  001442                                                            BEQ     50145$
                                    LET RECLOG :B= RECLOG + #1  ;SET RECORD LOGGED,
012342  105237  003501                                                    INCB    RECLOG
                                    IF CMDWRD EQ #RWD THEN      ;IF THIS IS A REWIND CMD:
012346  023727  003426  102010                                            CMP     CMDWRD,#RWD
012354  001003                                                            BNE     50146$
                                      LET RECCNT(R5) := #0      ;CLEAR RECORD COUNT,
012356  005065  003330                                                    CLR     RECCNT(R5)
                                    ELSE
012362  000431                                                            BR      50147$
012364                                                                  50146$:
                                      IF #BRF.C SETIN CMDWRD THEN       ;IF BRF USED, UPDATE RECORD COUNT.
012364  032737  004000  003426                                            BIT     #BRF.C,CMDWRD
012372  001425                                                            BEQ     50150$
                                        IF #MOD.CO NOTSETIN CMDWRD THEN  ;IF A FORWARD CMD:
012374  032737  000400  003426                                            BIT     #MOD.CO,CMDWRD
012402  001007                                                            BNE     50151$
                                          IF #MOD.CO NOTSETIN PCMDWD THEN  ;IF PREV CMD WAS A FWD ALSO:
012404  032737  000400  003432                                            BIT     #MOD.CO,PCMDWD
012412  001002                                                            BNE     50152$
                                            LET RECCNT(R5) := RECCNT(R5) + #1  ;INCREMENT RECORD COUNT.
012414  005265  003330                                                    INC     RECCNT(R5)
                                          ENDIF
012420                                                                  50152$:
                                        ELSE                        ;IF REVERSE CMD:
012420  000412                                                            BR      50153$
012422                                                                  50151$:
                                          IF #MOD.CO SETIN PCMDWD THEN  ;IF PREVIOUS CMD WAS A REV ALSO:
012422  032737  000400  003432                                            BIT     #MOD.CO,PCMDWD
012430  001406                                                            BEQ     50154$
                                            IF #X0.BOT NOTSETIN EOTFLG(R5) THEN ;WHEN NOT AT BOT THEN
012432  032765  000002  003516                                            BIT     #X0.BOT,EOTFLG(R5)
012440  001002                                                            BNE     50155$
                                              LET RECCNT(R5) := RECCNT(R5) - #1  ;DECREMENT RECORD COUNT.
012442  005365  003330                                                    DEC     RECCNT(R5)
                                            ENDIF
012446                                                                  50155$:
                                          ENDIF
012446                                                                  50154$:
                                        ENDIF
012446                                                                  50153$:
                                      ENDIF
                                    ENDIF
```

```
          012446                                                                  50150$:
     3532 012446                                  ENDIF
          012446                                                                  50147$:
     3533 012446                                ENDIF
          012446                                                                  50145$:
     3534 012446                                LET @DATAWT := RECCNT(R5)
          012446  016577  003330  170740                                          MOV     RECCNT(R5),@DATAWT
     3535 012454                              ENDIF
          012454                                                                  50144$:
     3536 012454  000207                      RTS     PC                  ;RETURN.
```

```
3538                                    ;       THIS IS THE ERROR CHECK SUBROUTINE.  AFTER INTERRUPT THIS
3539                                    ;       SUBROUTINE IS CALLED TO CHECK THE TK25 STATUS.
3540                                    ;       IF SPECIAL COND IS SET THEN THE TCC HANDLING SUBROUTINE IS ENTERED.
3541                                    ;       IF THE RFC IS NON ZERO FOR A COMMAND REQUIRING A BPCR,
3542                                    ;       THEN AN ERROR RFC IS REPORTED,
3543                                    ;       INPUTS:
3544                                    ;       OUTPUTS:
3545                                    ;       REGISTERS:      R2, R4.
3546                                    ;       CALLS:          TCC0-TCC7.
3547
3548 012456                            CHKERR::IF DEVTBL(R5) NE #NINUSE THEN
      012456  026527  002536  177774                                          CMP     DEVTBL(R5),#NINUSE
      012464  001570                                                          BEQ     50156$
3549 012466                                    IF #X1.EWN SETIN MSGPKT+MS.XS1 THEN     ;IF EARLY WARNING IS SET AND
      012466  032737  000010  002350                                          BIT     #X1.EWN,MSGPKT+MS.XS1
      012474  001461                                                          BEQ     50157$
3550 012476                                      IFB EWPRNT NE #0 THEN         ;IF END OF TRACK STATUS PRINTS ALLOWED
      012476  105737  002216                                                  TSTB    EWPRNT
      012502  001433                                                          BEQ     50160$
3551 012504                                      LET R2 := MSGPKT+MS.XS1 CLR.BY #177417
      012504  013702  002350                                                  MOV     MSGPKT+MS.XS1,R2
      012510  042702  177417                                                  BIC     #177417,R2
3552 012514                                      LET R2 := R2 SHIFT -4
      012514  006202                                                          ASR     R2
      012516  006202                                                          ASR     R2
      012520  006202                                                          ASR     R2
      012522  006202                                                          ASR     R2
3553 012524                                      PRINTF #EWMSG,R2              ;PRINT END OF TRACK MESSAGE
      012524  010246                                                          MOV     R2,-(SP)
      012526  012746  013050                                                  MOV     #EWMSG,-(SP)
      012532  012746  000002                                                  MOV     #2,-(SP)
      012536  010600                                                          MOV     SP,R0
      012540  104417                                                          TRAP    C$PNTF
      012542  062706  000006                                                  ADD     #6,SP
3554 012546                                    PRINTF  #TRKMSG,@DATAWT         ;*** TRACE
      012546  017746  170642                                                  MOV     @DATAWT,-(SP)
      012552  012746  013133                                                  MOV     #TRKMSG,-(SP)
      012556  012746  000002                                                  MOV     #2,-(SP)
      012562  010600                                                          MOV     SP,R0
      012564  104417                                                          TRAP    C$PNTF
      012566  062706  000006                                                  ADD     #6,SP
3555 012572                                    ENDIF
      012572                                                          50160$:
3556 012572                                    IF CTCC EQ #4 THEN              ;IF TCC4 AND WE DID A WRITE
      012572  023727  003456  000004                                          CMP     CTCC,#4
      012600  001017                                                          BNE     50161$
3557 012602                                      IF CMDWRD EQ #WRT OR CMDWRD EQ #WTM THEN
      012602  023727  003426  104005                                          CMP     CMDWRD,#WRT
      012610  001404                                                          BEQ     50162$
      012612  023727  003426  100011                                          CMP     CMDWRD,#WTM
      012620  001007                                                          BNE     50163$
      012622                                                          50162$:
3558 012622                                        IFB IREC EQ #0 THEN
      012622  105737  002210                                                  TSTB    IREC
      012626  001004                                                          BNE     50164$
3559 012630  004737  013174                      JSR       PC,EWRTRY         ;DO EW RETRY
3560 012634  000137  013046                      JMP       1$                ;ALAS, A "GOTO" STATEMENT, EH?
```

```
 3561 012640                                   ENDIF
      012640                                                                          5016 ;:
 3562 012640                                  ENDIF
      012640                                                                          50163$:
 3563 012640                                 ENDIF
      012640                                                                          50161$:
 3564 012640                                ENDIF
      012640                                                                          50157$:
 3565 012640                                IF #TS.SC SETIN TSSREG THEN       ;IF SPECIAL COND STATUS IS SET THEN:
      012640  032737  100000  003462                                                  BIT     #TS.SC,TSSREG
      012646  001441                                                                  BEQ     50165$
 3566 012650                                 IF CTCC NE #2 THEN               ;IF TCC IS NOT 2 THEN:
      012650  023727  003456  000002                                                  CMP     CTCC,#2
      012656  001405                                                                  BEQ     50166$
 3567 012660                                  IFB ERRREC EQ #0 THEN           ;IF NOT IN ERROR RECOVERY:
      012660  105737  003505                                                          TSTB    ERRREC
      012664  001002                                                                  BNE     50167$
 3568 012666  005265  003270                   INC       SCCNT(R5)            ;INC SC COUNTER.
 3569 012672                                  ENDIF
      012672                                                                          50167$:
 3570 012672                                 ENDIF
      012672                                                                          50166$:
 3571 012672                                 IF #TS.NXM SETIN TSSREG OR #TS.UPE SETIN TSSREG THEN  ;WHEN NON-EXISTANT MEMO
      012672  032737  004000  003462                                                  BIT     #TS.NXM,TSSREG
      012700  001004                                                                  BNE     50170$
      012702  032737  040000  003462                                                  BIT     #TS.UPE,TSSREG
      012710  001412                                                                  BEQ     50171$
      012712                                                                          50170$:
 3572 012712                                  IF #X2.OPM NOTSETIN MSGPKT+MS.XS2 THEN                   ;AND TAPE NOT MOVED
      012712  032737  100000  002352                                                  BIT     #X2.OPM,MSGPKT+MS.XS2
      012720  001003                                                                  BNE     50172$
 3573 012722                                   LET R2 := #5                   ;SET TCC5 INDEX
      012722  012702  000005                                                          MOV     #5,R2
 3574 012726                                  ELSE
      012726  000402                                                                  BR      50173$
      012730                                                                          50172$:
 3575 012730                                   LET R2 := #4                   ;TAPE MOVED, SET TCC4 INDEX
      012730  012702  000004                                                          MOV     #4,R2
 3576 012734                                  ENDIF
      012734                                                                          50173$:
 3577 012734                                 ELSE
      012734  000402                                                                  BR      50174$
      012736                                                                          50171$:
 3578 012736                                  LET R2 := CTCC                  ;SET DETECTED TCC INDEX
      012736  013702  003456                                                          MOV     CTCC,R2
 3579 012742                                 ENDIF
      012742                                                                          50174$:
 3580 012742                                 LET R2 := R2 SHIFT 1  ;CURRENT TCC X 2.
      012742  006302                                                                  ASL     R2
 3581 012744  004772  013154                  JSR   PC,@TCCRA(R2)              ;GO TO THE TCC HANDLING SUBROUTINE.
 3582 012750                                ELSE
      012750  000430                                                                  BR      50175$
      012752                                                                          50165$:
 3583 012752                                 IF #BRF.C SETIN CMDWRD THEN      ;IF BRF IS USED IN THIS CMD THEN:
      012752  032737  004000  003426                                                  BIT     #BRF.C,CMDWRD
      012760  001424                                                                  BEQ     50176$
 3584 012762                                  IF MSGPKT+MS.RFC NE #0 THEN     ;IF THERE IS AN RFC THEN:
```

```
                012762  005737  002344                                                  TST     MSGPKT+MS.RFC
                012766  001421                                                          BEQ     50177$
     3585       012770                          IFB RANDOM EQ #0 ORB VFYFLG NE #0 THEN
                012770  105737  003533                                                  TSTB    RANDOM
                012774  001403                                                          BEQ     50200$
                012776  105737  003534                                                  TSTB    VFYFLG
                013002  001413                                                          BEQ     50201$
                013004                                                                  50200$:
     3586                                                       ;IF NOT IN RANDOM OR IF CMD IS WTV:
     3587       013004                            IFB IRE EQ #0 THEN                    ;IF RFC ERROR REPORTS ARE ALLOWED:
                013004  105737  003540                                                  TSTB    IRE
                013010  001010                                                          BNE     50202$
     3588       013012                            LET HRDCNT(R5) := HRDCNT(R5) + #1     ;UPDATE HARD ERROR COUNT
                013012  005265  003310                                                  INC     HRDCNT(R5)
     3589       013016                            ERRHRD 13,RFCERM,STAERM ;REPORT RFC ERROR
                013016  104456                                                          TRAP    C$ERHRD
                013020  000015                                                          .WORD   13
                013022  004350                                                          .WORD   RFCERM
                013024  005640                                                          .WORD   STAERM
     3590       013026  004737  014436                    JSR     PC,RAMDUM             ;GO DO RAM DUMP
     3591       013032                                    ENDIF
                013032                                                                  50202$:
     3592       013032                                ENDIF
                013032                                                                  50201$:
     3593       013032                            ENDIF
                013032                                                                  50177$:
     3594       013032                        ENDIF
                013032                                                                  50176$:
     3595       013032                    ENDIF
                013032                                                                  50175$:
     3596       013032                    IFB RWERR NE #0 THEN                  ;IF A READ/WRITE ERROR HAS OCCURRED THEN:
                013032  105737  003503                                                  TSTB    RWERR
                013036  001403                                                          BEQ     50203$
     3597       013040                      LET CMDPKT := CMDSAV                  ;RESTORE CMD PACKET AFTER ERROR RECOV.
                013040  013737  003430  002314                                        MOV     CMDSAV,CMDPKT
     3598       013046                    ENDIF
                013046                                                                  50203$:
     3599       013046                    ENDIF
                013046                                                                  50156$:
     3600       013046  000207          1$:       RTS     PC                    ;RETURN.
     3601
     3602       013050      045     116     045  EWMSG:    .ASCIZ  /%N%AEARLY WARNING DURING WRITE AT END OF TRACK %D2/
                013053      101     105     101
                013056      122     114     131
                013061      040     127     101
                013064      122     116     111
                013067      116     107     040
                013072      104     125     122
                013075      111     116     107
                013100      040     127     122
                013103      111     124     105
                013106      040     101     124
                013111      040     105     116
                013114      104     040     117
                013117      106     040     124
                013122      122     101     103
                013125      113     040     045
```

```
        013130     104     062     000
 3603   013133     045     101     040  TRKMSG: .ASCIZ  /%A RECORD NO %D5/
        013136     122     105     103
        013141     117     122     104
        013144     040     116     117
        013147     040     045     104
        013152     065     000
 3604                                            .EVEN
 3605
 3606
 3607                                    ;       ADDRESSES OF TCC HANDLING ROUTINES FOR TERMINATION CLASS CODES 0 - 7.
 3608
 3609   013154  013306                   TCCRA:  TCC0
 3610   013156  013330                           TCC1
 3611   013160  013346                           TCC2
 3612   013162  013514                           TCC3
 3613   013164  013536                           TCC4
 3614   013166  014256                           TCC5
 3615   013170  014360                           TCC6
 3616   013172  014414                           TCC7
```

```
                                ;       SUBROUTINE TO HANDLE EARLY WARNING WRITE ERRORS.
                                ;
                                ;       THIS ROUTINE WILL SIMPLY TAKE CARE OF EARLY WARNING WRITE ERRORS
                                ;       BY REISSUING THE COMMAND WHICH FAILED WITH THE RETRY MODIFIER SET.
                                ;       NOTE THAT NO ERROR FLAGS, COUNTS, ETC ARE CHANGED.
                                ;       (EW ERRORS WILL NOT SHOW UP IN ERROR TALLIES).

013174                          EWRTRY::
                                        LET -(SP) := CMDPKT             ;SAVE THE COMMAND PACKET
013174  013746  002314                                                          MOV     CMDPKT,-(SP)
                                        LET CMDPKT := CMDPKT SET.BY #MOD.C1     ;SET THE RETRY MODIFIER IN COMMAND
013200  052737  001000  002314                                                  BIS     #MOD.C1,CMDPKT
                                        LET ERRREC :B= #1               ;SHOW ERROR RECOVERY IN PROCESS
013206  112737  000001  003505                                                  MOVB    #1,ERRREC
                                        JSR     PC,EXCUTE               ;DO THE COMMAND
013214  004737  010774
                                        LET ERRREC :B= #0               ;NO MORE ERROR RECOVERY
013220  105037  003505                                                          CLRB    ERRREC
                                        LET CMDPKT := (SP)+             ;RESTORE THE ORIGINAL COMMAND STATUS
013224  012637  002314                                                          MOV     (SP)+,CMDPKT
                                        LET R2 := (SP)+                 ;MODIFY THE RETURN ADDRESS
013230  012602                                                                  MOV     (SP)+,R2
                                        JSR     PC,WSSR                 ;WAIT FOR THE COMMAND TO FINISH
013232  004737  012124
                                        LET -(SP) := #10.               ;OVERALL LOOP CONTROL
013236  012746  000012                                                          MOV     #10.,-(SP)
                                        REPEAT                          ;LOOP
013242                                                                  50204$:
                                        DELAY 250.                      ;WAIT FOR THE UNIT TO STOP
013242  012727  000372                                                          MOV     #250.,(PC)+
013246  000000                                                                  .WORD   0
013250  013727  002116                                                          MOV     L$DLY,(PC)+
013254  000000                                                                  .WORD   0
013256  005367  177772                                                          DEC     -6(PC)
013262  001375                                                                  BNE     .-4
013264  005367  177756                                                          DEC     -22(PC)
013270  001367                                                                  BNE     .-20
                                        LET (SP) := (SP) - #1           ;ONE LESS ITERATION
013272  005316                                                                  DEC     (SP)
                                        UNTIL (SP) EQ #0                ;UNTIL = 0
013274  005716                                                                  TST     (SP)
013276  001361                                                                  BNE     50204$
                                        LET SP := SP + #2               ;CORRECT THE STACK
013300  062706  000002                                                          ADD     #2,SP
                                        RTS     PC                      ;RETURN TO CALLER
013304  000207
```

```
                                ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 0, UNDEFINED SPECIAL
                                ;       CONDITION ERROR.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:
                                
013306                          TCC0::  LET HRDCNT(R5) := HRDCNT(R5) + #1  ;UPDATE HARD ERROR COUNT.
013306  005265  003310                                                          INC     HRDCNT(R5)
013312                                  ERRHRD 5,SCERM,STAERM                   ;REPORT SPECIAL CONDITION ERROR.
013312  104456                                                                  TRAP    C$ERHRD
013314  000005                                                                  .WORD   5
013316  004324                                                                  .WORD   SCERM
013320  005640                                                                  .WORD   STAERM
013322  004737  014436                  JSR     PC,RAMDUM               ;GO DO RAM DUMP
013326  000207                          RTS PC                          ;RETURN.
                                
                                
                                
                                
                                ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 1, ATTENTION CONDITION.
                                ;       THIS TCC INDICATES THAT THE DRIVE HAS UNDERGONE A STATUS CHANGE
                                ;       SUCH AS GOING OFFLINE OR COMING ONLINE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2,R4
                                ;       CALLS:          DROPU
                                
013330                          TCC1::  ERRDF 6,ATTNM,STAERM            ;REPORT ATTENTION-UNIT OFF LINE.
013330  104455                                                                  TRAP    C$ERDF
013332  000006                                                                  .WORD   6
013334  004431                                                                  .WORD   ATTNM
013336  005640                                                                  .WORD   STAERM
013340  004737  014436                  JSR     PC,RAMDUM               ;GO DO RAM DUMP
013344  000207                          RTS     PC                      ;RETURN.
```

```
                                ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 2, TAPE STATUS ALERT.
                                ;       A STATUS CONDITION HAS BEEN ENCOUNTERED THAT MAY HAVE SIGNIFICANCE
                                ;       TO THE PROGRAM.  BITS OF INTEREST INCLUDE TMK, RLS, LET, RLL, BOT, EOT.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

013346                          TCC2::    IF #X0.BOT SETIN MSGPKT+MS.XS0 ANDB EXPBOT NE #0 THEN
013346  032737  000002  002346                                                  BIT     #X0.BOT,MSGPKT+MS.XS0
013354  001404                                                                  BEQ     50205$
013356  105737  003532                                                          TSTB    EXPBOT
013362  001401                                                                  BEQ     50205$
                                                                        ;IF AT BOT AND BOT IS EXPECTED:
013364  000452                    BR TC2RTN                               ;RETURN-TCC2 CAUSED BY EXPECTED BOT.
013366                          ENDIF
013366                                                                  50205$:
013366                          IF L$TEST EQ #3 THEN
013366  023727  002114  000003                                                  CMP     L$TEST,#3
013374  001011                                                                  BNE     50206$
013376                           IF PCMDWD EQ #WTM AND CMDWRD EQ #SRR THEN
013376  023727  003432  100011                                                  CMP     PCMDWD,#WTM
013404  001005                                                                  BNE     50207$
013406  023727  003426  104410                                                  CMP     CMDWRD,#SRR
013414  001001                                                                  BNE     50207$
013416  000435                    BR    TC2RTN
013420                           ENDIF
013420                                                                  50207$:
013420                          ENDIF
013420                                                                  50206$:
013420                          IF #X0.RLS!X0.RLL!X0.TMK!X0.LET!X0.BOT SETIN MSGPKT+MS.XS0 THEN
013420  032737  170002  002346                                                  BIT     #X0.RLS!X0.RLL!X0.TMK!X0.LET
013426  001431                                                                  BEQ     50210$
                                                                        ;IF TCC2 CAUSED BY ANYTHING BUT EOT:
                                  IFB RANDOM EQ #0 ORB VFYFLG NE #0 THEN
013430  105737  003533                                                          TSTB    RANDOM
013434  001403                                                                  BEQ     50211$
013436  105737  003534                                                          TSTB    VFYFLG
013442  001423                                                                  BEQ     50212$
013444                                                                  50211$:
                                                                        ;IF NOT IN RANDOM OR IF CMD IS WTV:
013444                            IFB IRE EQ #0 THEN                      ;IF RFC ERROR REPORTS ARE ALLOWED:
013444  105737  003540                                                          TSTB    IRE
013450  001020                                                                  BNE     50213$
013452                              IFB ERRREC NE #0 THEN                 ;IF WE ARE IN ERROR RECOVERY THEN:
013452  105737  003505                                                          TSTB    ERRREC
013456  001403                                                                  BEQ     50214$
013460                                LET UNREC :B= UNREC + #1            ;SET UNRECOVERABLE FLAG FOR LOG.
013460  105237  003504                                                          INCB    UNREC
013464                              ELSE                                  ;ELSE - IF NOT IN ERROR RECOVERY:
013464  000402                                                                  BR      50215$
013466                                                                  50214$:
013466                                LET SCCNT(R5) := SCCNT(R5) + #1 ;INCREMENT THE SPEC COND COUNTER.
013466  005265  003270                                                          INC     SCCNT(R5)
013472                              ENDIF
013472                                                                  50215$:
013472                                LET HRDCNT(R5) := HRDCNT(R5) + #1   ;UPDATE HARD ERROR COUNT.
```

```
        013472  005265  003310                                                  INC     HRDCNT(R5)
3698    013476                                  ERRHRD 7,TSAM,STAERM            ;REPORT TAPE STATUS ALERT.
        013476  104456                                                          TRAP    C$ERHRD
        013500  000007                                                          .WORD   7
        013502  004532                                                          .WORD   TSAM
        013504  005640                                                          .WORD   STAERM
3699    013506  004737  014436                    JSR     PC,RAMDUM             ;GO DO RAM DUMP
3700    013512                                  ENDIF
        013512                                                                  50213$:
3701    013512                                ENDIF
        013512                                                                  50212$:
3702    013512                              ENDIF
        013512                                                                  50210$:
3703    013512  000207          TC2RTN:     RTS PC                              ;RETURN.
3704
3705
3706
3707
3708
3709
3710                            ;       THE SPECIFIED FUNCTION WAS NOT INITIATED.  BITS OF INTEREST ARE
3711                            ;       RMR, OFL, VCK, BOT, ILC, WLE, ILA, AND NBA.
3712                            ;       INPUTS:
3713                            ;       OUTPUTS:
3714                            ;       REGISTERS:      R2,R4
3715                            ;       CALLS:          DROPU
3716
3717    013514                  TCC3::  ERRDF 8,FUNRM,STAERM                    ;REPORT FUNCTION REJECT.
        013514  104455                                                          TRAP    C$ERDF
        013516  000010                                                          .WORD   8
        013520  004447                                                          .WORD   FUNRM
        013522  005640                                                          .WORD   STAERM
3718    013524  004737  014436            JSR     PC,RAMDUM                   ;GO DO RAM DUMP
3719    013530  004737  017770            JSR     PC,DROPU                    ;DROP THE UNIT.
3720    013534  000207                    RTS PC                              ;RETURN.
```

```
                                ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 4, RECOVERABLE ERROR.
                                ;       TAPE POSITION IS ONE RECORD BEYOND WHAT ITS POSITION WAS WHEN
                                ;       THE FUNCTION WAS INITIATED.  RECOVERY PROCEDURE IS TO LOG THE
                                ;       ERROR AND ISSUE THE APPROPRIATE RETRY COMMAND.
                                ;       2 WRITE-ERROR-RECOVERY ALGORITHMS CAN BE SELECTED:
                                ;       THE FIRST ONE, VIA BADTSW SWITCH,  DOES DETECT BAD SPOTS ON TAPE.
                                ;       IT CALLS A WRITE RETRY SUBR UNTIL THE RECORD IS RECOVERED
                                ;       OR 20 BAD SPOTS HAVE BEEN LOGGED. AFTER LOGGING 256 BAD
                                ;       SPOTS, A BAD TAPE OVERFLOW MSG IS PRINTED AND THE
                                ;       UNIT DROPPED.
                                ;       THE SECOND ALGORITHM ISSUES THE TK25 WRITE RETRY COMMAND
                                ;       UP TO 16 TIMES BEFORE DROPPING THE UNIT OR PROCEEDING
                                ;       WITH THE NEXT RECORD ON RECOVERY.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2,R4.
                                ;       CALLS:          RTLE, EXCUTE, GOWAIT, DROPU, WRTY

013536                          TCC4::  IF DEVTBL(R5) EQ #NINUSE THEN
013536  026527  002536  177774                                    CMP     DEVTBL(R5),#NINUSE
013544  001002                                                    BNE     50216$
3741 013546  000137  014254            JMP     3$
3742 013552                            ENDIF
013552                                                          50216$:
3743 013552                            IF CMDLG EQ #2 ANDB BADTSW NE #0 THEN
013552  023727  003434  000002                                    CMP     CMDLG,#2
013560  001152                                                    BNE     50217$
013562  105737  002211                                            TSTB    BADTSW
013566  001547                                                    BEQ     50217$
3744 013570                    IFB ERRREC EQ #0 ANDB ERCVER NE #0 THEN
013570  105737  003505                                            TSTB    ERRREC
013574  001011                                                    BNE     50220$
013576  105737  002207                                            TSTB    ERCVER
013602  001406                                                    BEQ     50220$
3745 013604                    ERRSOFT 9,RERM,STAERM ;
013604  104457                                                    TRAP    C$ERSOFT
013606  000011                                                    .WORD   9
013610  004644                                                    .WORD   RERM
013612  005640                                                    .WORD   STAERM
3746 013614  004737  014436    JSR   PC,RAMDUM               ;GO DO RAM DUMP
3747 013620                    ENDIF
013620                                                          50220$:
3748 013620                    IFB IREC EQ #0 THEN             ;
013620  105737  002210                                            TSTB    IREC
013624  001125                                                    BNE     50221$
3749 013626                      LET ERRREC :B= ERRREC + #1  ;RETRY FLAG FOR EXCUTE SUBR: DON'T UPDATE REC CNT
013626  105237  003505                                            INCB    ERRREC
3750 013632                      LET WRTYER :B= WRTYER + #1  ;REWRITE ERROR FLAG FOR WRTY SUBR
013632  105237  003500                                            INCB    WRTYER
3751 013636                      IFB WRTYFG EQ #0 THEN       ;FIRST RETRY ON THIS RECORD: SUBSEQUENT
013636  105737  003477                                            TSTB    WRTYFG
013642  001115                                                    BNE     50222$
3752                                                            ;RETRIES WITH TCC4 ERRORS BY-PASS THIS SECTION
3753 013644                        LET WTYWRD := CMDWRD        ;SAVE WRITE COMMAND PACKET
013644  013737  003426  015316                                    MOV     CMDWRD,WTYWRD
3754 013652                        LET WTYCMD := CMDPKT        ;
013652  013737  002314  015314                                    MOV     CMDPKT,WTYCMD
```

H8


```
 013660
 013660  013737  002322  015320          LET WTYBRF := CMDPKT+CP.CNT           ;
                                                                        MOV     CMDPKT+CP.CNT,WTYBRF
 013666                                  LET RWERR :B= RWERR + #1  ;LOG SUBR FLAG: COUNT WRT ERRORS
 013666  105237  003503                                                 INCB    RWERR
 013672                                  LET WRTYFG :B= WRTYFG + #1         ;RETRY IN PROGRESS FLAG
 013672  105237  003477                                                 INCB    WRTYFG
 013676                                  REPEAT
 013676                                                                 50223$:
 013676                          1$:
 013676                                   LET WRTYCT(R5) := WRTYCT(R5) + #1     ;COUNT GLOBAL WRITE RETRIES
 013676  005265  003250                                                 INC     WRTYCT(R5)
 013702                                   LET RETRYC := #0                  ;CLEAR # OF RETRIES PER RECORD
 013702  005037  003474                                                 CLR     RETRYC
 013706                                   LET RPTCNT :B= #0                 ;CLEAR # OF REPEATS
 013706  105037  003476                                                 CLRB    RPTCNT
 013712  004737  014736                   JSR PC,WRTY              ;CALL WRITE RETRY
 013716                                           IF DEVTBL(R5) EQ #NINUSE THEN
 013716  026527  002536  177774                                         CMP     DEVTBL(R5),#NINUSE
 013724  001001                                                         BNE     50224$
 013726  000552                           BR 3$
 013730                                           ENDIF
 013730                                                                 50224$:
 013730                                  UNTILB WRTYER EQ #0 OR @BTPT HIS #40.   ;REPEAT RETRIES ON SAME RECORD
 013730  105737  003500                                                 TSTB    WRTYER
 013734  001404                                                         BEQ     50225$
 013736  027727  167564  000050                                         CMP     @BTPT,#40.
 013744  103754                                                         BLO     50223$
 013746                                                                 50225$:
                                                              ;UNTIL RECOVERED OR 20 BAD SPOTS
 013746                                IFB WRTYER NE #0 THEN
 013746  105737  003500                                                 TSTB    WRTYER
 013752  001442                                                         BEQ     50226$
 013754                                  IF @BTPT HIS #40. THEN     ;WHEN 20 BAD SPOTS LOGGED (TWR)
 013754  027727  167546  000050                                         CMP     @BTPT,#40.
 013762  103405                                                         BLO     50227$
 013764                                   LET TC4STO :B= ERCVER
 013764  113737  002207  003530                                         MOVB    ERCVER,TC4STO
 013772                                   LET ERCVER :B= #0          ;REMOVE THE MASK
 013772  105037  002207                                                 CLRB    ERCVER
 013776                                  ENDIF
 013776                                                                 50227$:
 013776                                  IF @BTPT HIS #512. THEN
 013776  027727  167524  001000                                         CMP     @BTPT,#512.
 014004  103423                                                         BLO     50230$
 014006                                    PRINTB  #BTMSG2         ;BAD TAPE OVERFLOW MESSAGE
 014006  012746  015407                                                         MOV     #BTMSG2,-(SP)
 014012  012746  000001                                                         MOV     #1,-(SP)
 014016  010600                                                                 MOV     SP,R0
 014020  104414                                                                 TRAP    C$PNTB
 014022  062706  000004                                                         ADD     #4,SP
 014026  004737  017770                    JSR     PC,DROPU         ;DROP THE UNIT.
 014032                                    LET RECCNT(R5) :=  #0    ;
 014032  005065  003330                                                 CLR     RECCNT(R5)
 014036                                    LET ERCVER :B= TC4STO    ;RESTORE THE MASK
 014036  113737  003530  002207                                         MOVB    TC4STO,ERCVER
 014044                                    LET @TSDB(R5) := #RWCPK ;REWIND UNIT
 014044  012775  002334  002456                                         MOV     #RWCPK,@TSDB(R5)
```

```
3780 014052
     014052  000402
     014054
3781 014054  000137  013676
3782 014060
     014060
3783 014060
     014060
3784 014060
     014060  105037  003477
3785 014064
     014064  105237  003551
3786 014070
     014070  013737  015316  003432
3787 014076
     014076
3788 014076
     014076  000402
     014100
3789 014100
     014100  105237  003504
3790 014104
     014104
3791 014104
     014104  000463
     014106
3792 014106  004737  014600
3793 014112
     014112  023727  003434  000002
     014120  003411
3794 014122
     014122  012702  000020
     014126  006202
3795 014130
     014130  023702  003474
     014134  002403
3796 014136
     014136  052737  020000  002314
3797 014144
     014144
3798 014144
     014144
3799 014144
     014144  005737  003474
     014150  001011
     014152  105737  002207
     014156  001406
3800 014160
     014160  104457
     014162  000011
     014164  004644
     014166  005640
3801 014170  004737  014436
3802 014174
     014174
3803 014174
     014174  005237  003474
```

```
        ELSE

          JMP     1$                ;LOOP
        ENDIF

      ENDIF

        LET WRTYFG :B= #0           ;RETRY COMPLETE FLAG
        LET MISCFG :B= MISCFG + #1          ;DO NOT HALT ON THIS CMD FLG
        LET PCMDWD := WTYWRD        ;RESTORE ORIGINAL WRT CMD AFTER RECOVERY
      ENDIF

    ELSE

      LET UNREC :B= UNREC + #1      ;
    ENDIF

  ELSE

    JSR PC,RTLE                     ;CHECK FOR RETRY LIMIT EXCEEDED.
    IF CMDLG GT #2 THEN             ;IF READ CMD THEN:

      LET R2 := #RRECL SHIFT -1     ;R2=READ RETRY COUNT LIMIT / 2

      IF RETRYC GE R2 THEN                  ;IF RETRY COUNT IS MORE THAN HALF LIMIT:

        LET CMDPKT := CMDPKT SET.BY #OPP.C  ;SET OPPOSITE BIT FOR RETRY2.
      ENDIF

    ENDIF

    IF RETRYC EQ #0 ANDB ERCVER NE #0 THEN  ;IF THIS IS THE ORIGINAL ERROR THEN:

    ERRSOFT 9,RERM,STAERM           ;REPORT RECOVERABLE ERROR

    JSR   PC,RAMDUM                 ;GO DO RAM DUMP
    ENDIF                           ;PROVIDED OPERATOR HAS ENABLED THE REPORT

    LET RETRYC := RETRYC + #1       ;UPDATE RETRY COUNT.
```

```
                BR      50231$
50230$:

                JMP     1$              ;LOOP

50231$:

50226$:
                CLRB    WRTYFG          ;RETRY COMPLETE FLAG
                INCB    MISCFG          ;DO NOT HALT ON THIS CMD FLG
                MOV     WTYWRD,PCMDWD   ;RESTORE ORIGINAL WRT CMD AFTER RECOVERY

50222$:

                BR      50232$
50221$:
                INCB    UNREC

50232$:

                BR      50233$
50217$:
                CMP     CMDLG,#2        ;IF READ CMD THEN:
                BLE     50234$
                MOV     #RRECL,R2       ;R2=READ RETRY COUNT LIMIT / 2
                ASR     R2
                CMP     RETRYC,R2       ;IF RETRY COUNT IS MORE THAN HALF LIMIT:
                BLT     50235$
                BIS     #OPP.C,CMDPKT   ;SET OPPOSITE BIT FOR RETRY2.

50235$:

50234$:
                TST     RETRYC          ;IF THIS IS THE ORIGINAL ERROR THEN:
                BNE     50236$
                TSTB    ERCVER
                BEQ     50236$
                TRAP    C$ERSOFT        ;REPORT RECOVERABLE ERROR
                .WORD   9
                .WORD   RERM
                .WORD   STAERM
                JSR     PC,RAMDUM       ;GO DO RAM DUMP
                                        ;PROVIDED OPERATOR HAS ENABLED THE REPORT
50236$:
                INC     RETRYC          ;UPDATE RETRY COUNT.
```

```
3804 014200
     014200  052737  001000  002314          LET CMDPKT := CMDPKT SET.BY #MOD.C1    ;SET RETRY BIT IN CMD PACKET.
                                                                                        BIS     #MOD.C1,CMDPKT
3805 014206
     014206  105737  002210                  IFB IREC EQ #0 THEN            ;IF ERROR RECOVERY ENABLED:
     014212  001016                                                                     TSTB    IREC
                                                                                        BNE     50237$
3806 014214                                           IF DEVTBL(R5) EQ #NINUSE THEN
     014214  026527  002536  177774                                                    CMP     DEVTBL(R5),#NINUSE
     014222  001001                                                                    BNE     50240$
3807 014224  000413                               BR 3$
3808 014226                                           ENDIF
     014226                                                                          50240$:
3809 014226                                     LET ERRREC :B= ERRREC + #1  ;SET ERROR RECOVERY FLAG.
     014226  105237  003505                                                          INCB    ERRREC
3810 014232                                     POP R2,R2                  ;POP 2 RTN ADRS FROM STACK.
     014232  012602                                                                 MOV     (SP)+,R2
     014234  012602                                                                 MOV     (SP)+,R2
3811 014236  004737  010774                     JSR PC,EXCUTE                   ;GO EXECUTE THE RETRY COMMAND.
3812 014242  000137  011470                     JMP GOWAIT                 ;GO WAIT FOR INTERRUPT + CHECK STATUS.
3813 014246                                     ELSE                       ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
     014246  000402                                                                 BR      50241$
     014250                                                                     50237$:
3814 014250                                     LET UNREC :B= UNREC + #1    ;SET UNRECOVERABLE ERROR FLAG.
     014250  105237  003504                                                          INCB    UNREC
3815 014254                                     ENDIF
     014254                                                                     50241$:
3816 014254                                ENDIF
     014254                                                                     50233$:
3817 014254  000207                  3$:   RTS PC                             ;RETURN
```

```
                                ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 5, RECOVERABLE ERROR.
                                ;       TAPE POSITION HAS NOT CHANGED.  RECOVERY PROCEDURE IS TO LOG THE
                                ;       ERROR AND RE-ISSUE THE ORIGINAL COMMAND.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2,R4.
                                ;       CALLS:          RTLE, EXCUTE, GOWAIT, DROPU.

014256  004737  014600        TCC5::  JSR PC,RTLE                     ;CHECK FOR RETRY LIMIT EXCEEDED
                                        IF RETRYC EQ #0 THEN            ;IF THIS IS THE ORIGINAL ERROR THEN:
014262  005737  003474                                                          TST     RETRYC
014266  001006                                                                BNE     50242$
                                          ERRSOFT 10,RERM,STAERM        ;REPORT RECOVERABLE ERROR.
014270  104457                                                                  TRAP    C$ERSOFT
014272  000012                                                                  .WORD   10
014274  004644                                                                  .WORD   RERM
014276  005640                                                                  .WORD   STAERM
014300  004737  014436                  JSR     PC,RAMDUM               ;GO DO RAM DUMP
                                        ENDIF
014304                                                                  50242$:
                                        LET RETRYC := RETRYC + #1        ;UPDATE RETRY COUNTER.
014304  005237  003474                                                          INC     RETRYC
                                        IFB IREC EQ #0 THEN             ;IF ERROR RECOVERY IS ENABLED:
014310  105737  002210                                                          TSTB    IREC
014314  001016                                                                BNE     50243$
                                          LET ERRREC :B= ERRREC + #1     ;SET ERROR RECOVERY FLAG.
014316  105237  003505                                                          INCB    ERRREC
                                          LET RECCNT(R5) := RECCNT(R5) + #1      ;UPDATE REC COUNT
014322  005265  003330                                                          INC     RECCNT(R5)
                                          LET @DATAWT := RECCNT(R5)              ;AND INSERT IT INTO WRT BFR
014326  016577  003330  167060                                                  MOV     RECCNT(R5),@DATAWT
                                          POP R2,R2                     ;POP 2 RTN ADRS FROM STACK.
014334  012602                                                                  MOV     (SP)+,R2
014336  012602                                                                  MOV     (SP)+,R2
014340  004737  010774                    JSR PC,EXCUTE                 ;GO RE-ISSUE THE COMMAND.
014344  000137  011470                    JMP GOWAIT                    ;GO WAIT FOR INTERRUPT + CHECK STATUS.
                                        ELSE                            ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
014350  000402                                                                  BR      50244$
014352                                                                  50243$:
                                          LET UNREC :B= UNREC + #1       ;SET UNRECOVERABLE ERROR FLAG.
014352  105237  003504                                                          INCB    UNREC
                                        ENDIF
014356                                                                  50244$:
014356  000207                        RTS PC                          ;RETURN.
```

```
                                ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 6, UNRECOVERABLE ERROR.
                                ;       TAPE POSITION HAS BEEN LOST.  THE ONLY VALID RECOVERY PROCEDURE
                                ;       IS TO REWIND AND START OVER AT BOT UNLESS THE TAPE HAS LABELS OR
                                ;       SEQUENCE NUMBERS.  THIS DIAGNOSTIC WILL REWIND AND RETRY THE
                                ;       COMMAND ONLY IF DENSITY CHECK IS SET, OTHERWISE THE UNIT WILL BE
                                ;       DROPPED FROM THE TEST SEQUENCE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R4
                                ;       CALLS:          RTLE, WSSR, EXCUTE, GOWAIT, DROPU

014360                          TCC6::  LET @TSDB(R5) := #RWCPK         ;ISSUE A REWIND COMMAND,
014360  012775  002334  002456                                                  MOV     #RWCPK,@TSDB(R5)
014366  004737  012124                  JSR PC,WSSR                     ;WAIT FOR SUBSYSTEM READY,
014372                                  ERRDF 11,URERM,STAERM           ;REPORT UNRECOVERABLE ERROR.
014372  104455                                                                  TRAP    C$ERDF
014374  000013                                                                  .WORD   11
014376  004666                                                                  .WORD   URERM
014400  005640                                                                  .WORD   STAERM
014402  004737  014436                  JSR     PC,RAMDUM               ;GO DO RAM DUMP
014406  004737  017770                  JSR     PC,DROPU                ;DROP THE UNIT.
014412  000207                          RTS PC                          ;RETURN
```

```
                        ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 7, FATAL SUBSYSTEM
                        ;       ERROR.  THE SUBSYSTEM IS INCAPABLE OF PROPERLY PERFORMING
                        ;       COMMANDS OR AT LEAST ITS INTEGRITY IS SERIOUSLY QUESTIONABLE.
                        ;       REFER TO THE FATAL CLASS CODE FIELD IN THE TSSR REGISTER FOR
                        ;       ADDITIONAL INFORMATION ON THE TYPE OF FATAL ERROR.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2, R4
                        ;       CALLS:

014414                  TCC7::  ERRDF 12,FATSM,STAERM           ;REPORT FATAL SUBSYSTEM ERROR.
014414  104455                                                            TRAP    C$ERDF
014416  000014                                                            .WORD   12
014420  004467                                                            .WORD   FATSM
014422  005640                                                            .WORD   STAERM
014424  004737  014436          JSR     PC,RAMDUM               ;GO DO RAM DUMP
014430  004737  017770          JSR     PC,DROPU                ;DROP THE UNIT.
014434  000207                  RTS PC                          ;RETURN.


014436                  RAMDUM::
014436                          IFB RAMWRT NE #0 THEN
014436  105737  002214                                                    TSTB    RAMWRT
014442  001455                                                            BEQ     50245$
014444                          PRINTX  #RAMFHR
014444  012746  005320                                                    MOV     #RAMFHR,-(SP)
014450  012746  000001                                                    MOV     #1,-(SP)
014454  010600                                                            MOV     SP,R0
014456  104415                                                            TRAP    C$PNTX
014460  062706  000004                                                    ADD     #4,SP
014464  012737  000010  003406          MOV     #8.,RAMSIZ      ;RAM FIELD IS 8 BYTES LONG
014472  012737  000020  003342          MOV     #20,RAMHLD      ;COMMAND PACKET ADDRESS
014500  004737  016324                  JSR     PC,RAMER        ;READ AND PRINT THEM
014504  012737  000200  003342          MOV     #200,RAMHLD     ;CHARACTERISTICS PACKET ADDRESS
014512  004737  016324                  JSR     PC,RAMER        ;READ AND PRINT THEM
014516  012737  000016  003406          MOV     #14.,RAMSIZ     ;RAM FIELD IS 8 BYTES LONG
014524  012737  000214  003342          MOV     #214,RAMHLD     ;MESSAGE PACKET ADDRESS
014532  004737  016324                  JSR     PC,RAMER        ;READ AND PRINT THEM
014536  012737  000020  003406          MOV     #16.,RAMSIZ     ;RAM FIELD IS SIXTEEN BYTES LONG
014544  012737  000060  003342          MOV     #60,RAMHLD      ;SENCE BYTES ADDRESS
014552  004737  016324                  JSR     PC,RAMER        ;READ AND PRINT THEM
014556                  PRINTX  #RAMLIN
014556  012746  005462                                                    MOV     #RAMLIN,-(SP)
014562  012746  000001                                                    MOV     #1,-(SP)
014566  010600                                                            MOV     SP,R0
014570  104415                                                            TRAP    C$PNTX
014572  062706  000004                                                    ADD     #4,SP
014576                  ENDIF
014576                                                          50245$:
014576  000207                  RTS     PC
```

```
                                ;       SUBROUTINE TO CHECK FOR RETRY LIMIT EXCEEDED.  PRINTS ERROR MESSAGE
                                ;       IF EXCEEDED AND DROP UNIT UNLESS COMMAND IS A READ.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R4.
                                ;       CALLS:          DROPU
                                
014600                          RTLE::  IF CMDLG EQ #0 THEN             ;IF CMD IS NOT A READ OR WRITE THEN:
014600  005737  003434                                                           TST     CMDLG
014604  001012                                                            BNE     50246$
014606                                  ERRDF 11,URERM,STAERM           ;REPORT UNRECOVERABLE ERROR.
014606  104455                                                                   TRAP    C$ERDF
014610  000013                                                                  .WORD   11
014612  004666                                                                  .WORD   URERM
014614  005640                                                                  .WORD   STAERM
014616  004737  014436                  JSR     PC,RAMDUM               ;GO DO RAM DUMP
014622  004737  017770                  JSR     PC,DROPU                ;DROP THE UNIT.
014626                                  POP R2
014626  012602                                                                  MOV     (SP)+,R2
014630  000441                          BR RTLRTN                       ;AND RETURN.
014632                                  ENDIF
014632                                                                  50246$:
014632                                  LET RWERR :B= RWERR + #1         ;SET READ/WRITE ERROR FLAG.
014632  105237  003503                                                           INCB    RWERR
014636                                  IF CMDLG EQ #2 THEN             ;IF CMD IS A WRT OR WTM:
014636  023727  003434  000002                                                   CMP     CMDLG,#2
014644  001020                                                            BNE     50247$
014646                                    IF RETRYC EQ #WRECL THEN      ;IF RETRY COUNT HAS REACHED LIMIT:
014646  023727  003474  000020                                                   CMP     RETRYC,#WRECL
014654  001013                                                            BNE     50250$
014656                                    LET UNREC :B= UNREC + #1      ;SET UNRECOVERABLE FLAG
014656  105237  003504                                                           INCB    UNREC
014662                                    ERRDF 14,RLEXM,STAERM         ;REPORT RETRY LIMIT EXCEEDED.
014662  104455                                                                   TRAP    C$ERDF
014664  000016                                                                  .WORD   14
014666  004404                                                                  .WORD   RLEXM
014670  005640                                                                  .WORD   STAERM
014672  004737  014436                    JSR  PC,RAMDUM                ;GO DO RAM DUMP
014676  004737  017770                    JSR  PC,DROPU                 ;DROP THE UNIT.
014702                                    POP  R2
014702  012602                                                                  MOV     (SP)+,R2
014704                                  ENDIF
014704                                                                  50250$:
014704                                  ELSE                            ;ELSE - CMD IS A READ:
014704  000413                                                                  BR      50251$
014706                                                                  50247$:
014706                                    IF RETRYC EQ #RRECL THEN      ;IF RETRY COUNT HAS REACHED LIMIT:
014706  023727  003474  000020                                                   CMP     RETRYC,#RRECL
014714  001007                                                            BNE     50252$
014716                                      LET UNREC :B= UNREC + #1    ;SET UNRECOVERABLE FLAG
014716  105237  003504                                                           INCB    UNREC
014722                                      ERRHRD 14,RLEXM,STAERM      ;REPORT RECOVERABLE ERROR.
014722  104456                                                                   TRAP    C$ERHRD
014724  000016                                                                  .WORD   14
014726  004404                                                                  .WORD   RLEXM
014730  005640                                                                  .WORD   STAERM
```

```
014732                                    POP R2
014732  012602                                                                MOV     (SP)+,R2
014734                                  ENDIF
014734                                                                        50252$:
014734                                ENDIF
014734                                                                        50251$:
014734  000207                RTLRTN: RTS PC                       ;RETURN

                              ;       SUBR TO REWRITE A BAD, BUT RECOVERABLE WRITTEN RECORD.
                              ;       REWRITE RECORD ON SAME SPOT: REPEAT 4 TIMES.
                              ;       IF ALL 4 REPEATS GOOD, RECORD IS RECOVERED
                              ;       AND A RECOVERABLE WRITE ERROR IS LOGGED.
                              ;       IF ANY OF 4 REPEATS BAD, ERASE BAD RECORD, LOG SUSPECTED
                              ;       BAD SPOT, RETRY AGAIN. RETRY 4 TIMES, UP TO 4 REPEATS EACH.
                              ;       IF RECORD NOT GOOD AFTER 4 RETRIES, ERASE IT, EXIT WITH
                              ;       ERROR FLAG WRTYER SET, PRINTING RETRY FAILED.
                              ;       THIS ALL SCHEME IS REENTERED 20 TIMES MAX, IE 20 BAD
                              ;       SPOTS MAX ARE ALLOWED.

                              ;       INPUTS:
                              ;       OUTPUTS:
                              ;       REGISTERS:      R3,R4
                              ;       CALLS:          BORERS, REWRT

014736                        WRTY:: IF DEVTBL(R5) NE #NINUSE THEN        ;IF DRIVE NOT DROPPED
014736  026527  002536  177774                                        CMP     DEVTBL(R5),#NINUSE
014744  001537                                                        BEQ     50253$
014746                                BEGIN RETRY
014746                                  REPEAT
014746                                                                        50255$:
014746                                    BEGIN REPEAT
014746                                      REPEAT
014746                                                                        50257$:
014746  004737  015534                        JSR PC,BORERS             ;BACKSPACE/ERASE ONE RECORD
014752                                        LET WRTYER :B= #0         ;CLEAR WRITE RETRY ERROR
014752  105037  003500                                                CLRB    WRTYER
014756  004737  015710                        JSR PC,REWRT              ;REWRITE RECORD ON SAME SPOT
014762                                        IF DEVTBL(R5) EQ #NINUSE THEN
014762  026527  002536  177774                                        CMP     DEVTBL(R5),#NINUSE
014770  001003                                                        BNE     50260$
014772                                        LET RPTCNT :B= #3
014772  112737  000003  003476                                        MOVB    #3,RPTCNT
015000                                        ENDIF
015000                                                                        50260$:
015000                                        LET RPTCNT :B= RPTCNT + #1       ;COUNT REPEATS
015000  105237  003476                                                INCB    RPTCNT
015004                                      UNTILB RPTCNT EQ #4 ORB WRTYER NE #0   ;LIMIT: 4 REPEATS OR RECOVERED
015004  123727  003476  000004                                        CMPB    RPTCNT,#4
015012  001403                                                        BEQ     50261$
015014  105737  003500                                                TSTB    WRTYER
015020  001752                                                        BEQ     50257$
015022                                                                        50261$:
015022                                    END REPEAT                  ;
015022                                                                        50256$:
015022                                    LET RETRYC := RETRYC + #1        ;COUNT RETRIES
015022  005237  003474                                                INC     RETRYC
015026                                        IF DEVTBL(R5) EQ #NINUSE THEN
```

```
     015026  026527  002536  177774                                                        CMP     DEVTBL(R5),#NINUSE
     015034  001002                                                                        BNE     50262$
     015036  000502                         BR      3$
     015040                                 ELSE
     015040  000400                                                                        BR      50263$
     015042                                                                                50262$:
     015042                                 ENDIF
     015042                                                                                50263$:
     015042                          IFB WRTYER EQ #0 THEN                         ;
     015042  105737  003500                                                                TSTB    WRTYER
     015046  001001                                                                        BNE     50264$
     015050                            LEAVE RETRY                                 ;EXIT RETRY LOOP IF RECOVERED
     015050  000457                                                                        BR      50254$
     015052                          ELSE                                          ;
     015052                                                                                50264$:
     015052                            IFB ERCVER NE #0 THEN                       ;
     015052  105737  002207                                                                TSTB    ERCVER
     015056  001415                                                                        BEQ     50266$
     015060                              PRINTB #BTMSG1,RETRYC,<B,RPTCNT>                  ;PRINT SUSPECTED BAD SPOT
     015060  005046                                                                                CLR     -(SP)
     015062  153716  003476                                                                        BISB    RPTCNT,(SP)
     015066  013746  003474                                                                        MOV     RETRYC,-(SP)
     015072  012746  015322                                                                        MOV     #BTMSG1,-(SP)
     015076  012746  000003                                                                        MOV     #3,-(SP)
     015102  010600                                                                                MOV     SP,R0
     015104  104414                                                                                TRAP    C$PNTB
     015106  062706  000010                                                                        ADD     #10,SP
     015112                            ENDIF                                       ;
     015112                                                                                50266$:
     015112                            IF RETRYC EQ #1 THEN        ;ON FIRST RETRY, LOGG BAD SPOT
     015112  023727  003474  000001                                                        CMP     RETRYC,#1
     015120  001021                                                                        BNE     50267$
     015122                              LET BTPT := BTADDR(R5)                    ;BTPT IS BOTH THE BAD SPOT COUNTER
     015122  016537  002550  003526                                                        MOV     BTADDR(R5),BTPT
     015130                              LET R4 := @BTPT + #2                      ;AND THE LOGGING INDEX
     015130  017704  166372                                                                MOV     @BTPT,R4
     015134  062704  000002                                                                ADD     #2,R4
     015140                              LET @BTPT := R4                           ;
     015140  010477  166362                                                                MOV     R4,@BTPT
     015144                              IF R4 LOS #40. THEN                       ;
     015144  020427  000050                                                                CMP     R4,#40.
     015150  101005                                                                        BHI     50270$
     015152                                LET R3 := BTPT                          ;STORE FIRST 20 BAD SPOTS
     015152  013703  003526                                                                MOV     BTPT,R3
     015156                                LET R4 := R4 + R3                       ;
     015156  060304                                                                        ADD     R3,R4
     015160                                LET (R4) := RECCNT(R5)                  ;
     015160  016514  003330                                                                MOV     RECCNT(R5),(R4)
     015164                              ENDIF
     015164                                                                                50270$:
     015164                            ENDIF                                       ;
     015164                                                                                50267$:
     015164                            LET ERSFLG :B= ERSFLG + #1                  ;ERASE FLAG TO ERASE BAD RECORD
     015164  105237  003545                                                                INCB    ERSFLG
     015170                            LET RWERR :B= #0                            ;CANCELL "LOG" ERROR FLAG ON FAILING RETRY
     015170  105037  003503                                                                CLRB    RWERR
     015174                            LET RPTCNT :B= #0                           ;CLEAR REPEAT COUNT FOR NEXT RETRY
```

```
015174  105037  003476                                                        CLRB    [illegible]TCNT
015200                              ENDIF                            ;
015200                                                                        50265$:
015200                            UNTIL RETRYC EQ #4                 ;LIMIT:  4 RETRIES
015200  023727  003474  000004                                                CMP     RETRYC,#4
015206  001257                                                                BNE     50255$
015210                          END RETRY                            ;
015210                                                                        50254$:
015210                          IFB WRTYER NE #0 THEN                ;
015210  105737  003500                                                        TSTB    WRTYER
015214  001413                                                                BEQ     50271$
015216                            IFB ERCVER NE #0 THEN              ;
015216  105737  002207                                                        TSTB    ERCVER
015222  001410                                                                BEQ     50272$
015224                              PRINTB #BTMSG3                   ;PRINT RETRY FAILED
015224  012746  015464                                                                MOV     #BTMSG3,-(SP)
015230  012746  000001                                                                MOV     #1,-(SP)
015234  010600                                                                        MOV     SP,R0
015236  104414                                                                        TRAP    C$PNTB
015240  062706  000004                                                                ADD     #4,SP
015244                            ENDIF
015244                                                                        50272$:
015244                          ENDIF
015244                                                                        50271$:
015244                        ENDIF
015244                                                                        50253$:
015244                  3$:     LET -(SP) := #10.           ;LOOP CONTROL
015244  012746  000012                                                        MOV     #10.,-(SP)
015250                          REPEAT
015250                                                                        50273$:
015250                          DELAY   250.                ;WAIT FOR THE UNIT TO STOP
015250  012727  000372                                                                MOV     #250.,(PC)+
015254  000000                                                                        .WORD   0
015256  013727  002116                                                                MOV     L$DLY,(PC)+
015262  000000                                                                        .WORD   0
015264  005367  177772                                                                DEC     -6(PC)
015270  001375                                                                        BNE     .-4
015272  005367  177756                                                                DEC     -22(PC)
015276  001367                                                                        BNE     .-20
015300                          LET (SP) := (SP) - #1       ;ONE LESS ITERATION
015300  005316                                                                DEC     (SP)
015302                          UNTIL (SP) EQ #0            ;'TILL ZERO!
015302  005716                                                                TST     (SP)
015304  001361                                                                BNE     50273$
015306                          LET SP := SP + #2           ;POP THE STACK
015306  062706  000002                                                        ADD     #2,SP
015312  000207                  RTS PC




015314  000000          WTYCMD: .WORD   0       ;STORAGE FOR WRITE CMD WHILE RETRYING
015316  000000          WTYWRD: .WORD   0       ;STORAGE FOR WRITE CMD WORD WHILE RETRYING
015320  000000          WTYBRF: .WORD   0       ;STORAGE FOR WRITE BPCR WHILE RETRYING
```

```
 4015
 4016 015322  045  101  123  BTMSG1: .ASCIZ  /%ASUSPECT BAD SPOT AFTER %D1%A RETRY, %D1%A REPEAT%N/
      015325  125  123  120
      015330  105  103  124
      015333  040  102  101
      015336  104  040  123
      015341  120  117  124
      015344  040  101  106
      015347  124  105  122
      015352  040  045  104
      015355  061  045  101
      015360  040  122  105
      015363  124  122  131
      015366  054  040  045
      015371  104  061  045
      015374  101  040  122
      015377  105  120  105
      015402  101  124  045
      015405  116  000
 4017 015407  045  116  045  BTMSG2: .ASCIZ  /%N%ABAD TAPE OVERFLOW: CHANGE CARTRIDGE!%N%N/
      015412  101  102  101
      015415  104  040  124
      015420  101  120  105
      015423  040  117  126
      015426  105  122  106
      015431  114  117  127
      015434  072  040  103
      015437  110  101  116
      015442  107  105  040
      015445  103  101  122
      015450  124  122  111
      015453  104  107  105
      015456  041  045  116
      015461  045  116  000
 4018 015464  045  101  122  BTMSG3: .ASCIZ  /%ARETRY FAILED ON BAD SPOT...ERASED!%N/
      015467  105  124  122
      015472  131  040  106
      015475  101  111  114
      015500  105  104  040
      015503  117  116  040
      015506  102  101  104
      015511  040  123  120
      015514  117  124  056
      015517  056  056  105
      015522  122  101  123
      015525  105  104  041
      015530  045  116  000
 4019                          .EVEN
```

```
                                ;       SUBR TO BACSPACE ONE RECORD
                                ;       IF THE ERASE FLAG IS SET, THEN ERASE THAT RECORD
                                ;       INPUTS:         ERSFLG  1 = DO ERASE
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:          EXCUTE, GOWAIT, CKHAE

015534                          BORERS:: LET PCMDWD := CMDWRD   ;SET COMMAND  TO SPACE REV
015534  013737  003426  003432                                                  MOV     CMDWRD,PCMDWD
                                        LET CMDWRD := #SRR      ;
015542  012737  104410  003426                                                  MOV     #SRR,CMDWRD
                                        LET CMDPKT := CMDWRD CLR.BY #BRF.C      ;
015550  013737  003426  002314                                                  MOV     CMDWRD,CMDPKT
015556  042737  004000  002314                                                  BIC     #BRF.C,CMDPKT
                                        LET CMDSAV := CMDPKT    ;
015564  013737  002314  003430                                                  MOV     CMDPKT,CMDSAV
                                        LET CMDPKT+CP.ADL := #1 ;
015572  012737  000001  002316                                                  MOV     #1,CMDPKT+CP.ADL
                                        LET CMDLG := #0         ;
015600  005037  003434                                                          CLR     CMDLG
004737  007672                        JSR PC,CMDAC            ;
004737  010774                        JSR PC,EXCUTE           ;
004737  011470                        JSR PC,GOWAIT           ;
004737  020274                        JSR PC,CKHAE            ;
015624                                  IFB ERSFLG NE #0 THEN   ;WHEN ERASE FLAG IS SET, DO ERASE
105737  003545                                                                  TSTB    ERSFLG
001426                                                                          BEQ     50274$
015632                                    LET PCMDWD := CMDWRD  ;
013737  003426  003432                                                          MOV     CMDWRD,PCMDWD
015640                                    LET CMDWRD := #ERS    ;
012737  100411  003426                                                          MOV     #ERS,CMDWRD
015646                                    LET CMDPKT := CMDWRD  ;
013737  003426  002314                                                          MOV     CMDWRD,CMDPKT
015654                                    LET CMDSAV := CMDPKT  ;
013737  002314  003430                                                          MOV     CMDPKT,CMDSAV
004737  007672                          JSR PC,CMDAC            ;
004737  010774                          JSR PC,EXCUTE
004737  011470                          JSR PC,GOWAIT
004737  020274                          JSR PC,CKHAE
015702                                    LET ERSFLG :B= #0
105037  003545                                                                  CLRB    ERSFLG
                                        ENDIF
015706                                                                          50274$:
000207                                  RTS PC
                                ;       SUBR TO REWRITE A BADLY WRITTEN RECORD

015710                          REWRT:  IF DEVTBL(R5) NE #NINUSE THEN   ;IF DRIVE NOT DROPPED
026527  002536  177774                                                          CMP     DEVTBL(R5),#NINUSE
001441                                                                          BEQ     50275$
                                        LET PCMDWD := CMDWRD    ;RESTORE WRITE COMMAND PACKET
013737  003426  003432                                                          MOV     CMDWRD,PCMDWD
                                        LET CMDWRD := WTYWRD    ;
013737  015316  003426                                                          MOV     WTYWRD,CMDWRD
                                        LET CMDPKT := WTYCMD    ;
013737  015314  002314                                                          MOV     WTYCMD,CMDPKT
                                        LET CMDSAV := CMDPKT    ;
013737  002314  003430                                                          MOV     CMDPKT,CMDSAV
```

```
4057 015750
     015750  013737  003414  002316          LET CMDPKT+CP.ADL := DATAWT      ;
                                                        MOV     DATAWT,CMDPKT+CP.ADL
4058 015756
     015756  013737  015320  002322          LET CMDPKT+CP.CNT := WTYBRF      ;
                                                        MOV     WTYBRF,CMDPKT+CP.CNT
4059 015764
     015764  012737  000002  003434          LET CMDLG := #2                  ;
                                                        MOV     #2,CMDLG
4060 015772  004737  007672                  JSR PC,CMDAC
4061 015776  004737  010774                  JSR PC,EXCUTE                    ;RE-WRITE RECORD
4062 016002                                    IF DEVTBL(R5) NE #NINUSE THEN
     016002  026527  002536  177774                    CMP     DEVTBL(R5),#NINUSE
     016010  001404                                    BEQ     50276$
4063 016012  004737  011470                  JSR PC,GOWAIT                    ;
4064 016016  004737  020274                  JSR PC,CKHAE                     ;
4065 016022                                    ENDIF
     016022                                             50276$:
4066 016022                                  ENDIF
     016022                                             50275$:
4067 016022  000207                        RTS PC
```

```
                                ;       SUBROUTINE TO LOG BYTES READ/WRITTEN.
                                ;       ALSO UPDATES READ/WRITE ERROR COUNTERS.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R3, R4.
                                ;       CALLS:

4076 016024                     LOG::   IFB ERLOG EQ #0 THEN            ;IF DATA AND ERRORS HAVE NOT BEEN LOGGED THEN:
     016024  105737  003502                                                     TSTB    ERLOG
     016030  001126                                                             BNE     50277$
4077 016032                             LET ERLOG :B= ERLOG + #1         ;SET LOG DONE FLAG.
     016032  105237  003502                                                     INCB    ERLOG
4078 016036                             LET R4 := CMDLG                  ;GET CURRENT CMD LOGGING CODE.
     016036  013704  003434                                                     MOV     CMDLG,R4
4079 016042                             IF R4 NE #0 THEN                 ;IF THERE IS A CODE THEN:
     016042  005704                                                             TST     R4
     016044  001520                                                             BEQ     50300$
4080 016046                               LET R4 := R4 - #2              ;ADJUST THE CODE FOR TABLE INDEX.
     016046  162704  000002                                                     SUB     #2,R4
4081 016052                               LET R2 := R5 + BINC(R4) + #CNTBGN  ;R2 = ADR OF BYTE COUNT LSW.
     016052  010502                                                             MOV     R5,R2
     016054  066402  016310                                                     ADD     BINC(R4),R2
     016060  062702  002560                                                     ADD     #CNTBGN,R2
4082 016064                               LET (R2) := (R2) + BRFCNT      ;ADD BRF TO LSW.
     016064  063712  003424                                                     ADD     BRFCNT,(R2)
4083 016070                               IF MSGPKT+MS.RFC LOS BRFCNT THEN  ;IF THE RFC IS LOWER OR THE SAME AS BRF THEN:
     016070  023737  002344  003424                                             CMP     MSGPKT+MS.RFC,BRFCNT
     016076  101002                                                             BHI     50301$
4084 016100                                 LET (R2) := (R2) - MSGPKT+MS.RFC  ;SUBTRACT RFC FROM EXPECTED BRF.
     016100  163712  002344                                                     SUB     MSGPKT+MS.RFC,(R2)
4085 016104                               ENDIF
     016104                                                                     50301$:
4086 016104                             LET R3 := R2 + #10               ;R3 = ADR OF 2ND WORD.
     016104  010203                                                             MOV     R2,R3
     016106  062703  000010                                                     ADD     #10,R3
4087 016112                             WHILE (R2) GT #999. DO
     016112                                                                     50302$:
     016112  021227  001747                                                     CMP     (R2),#999.
     016116  003404                                                             BLE     50303$
4088 016120                               LET (R2) := (R2) - #1000.      ;UPDATE BYTE COUNT
     016120  162712  001750                                                     SUB     #1000.,(R2)
4089 016124                               LET (R3) := (R3) + #1          ;2ND WORD.
     016124  005213                                                             INC     (R3)
4090 016126                             ENDDO
     016126  000771                                                             BR      50302$
     016130                                                                     50303$:
4091 016130                             LET R2 := R3 + #10               ;R2 = ADR OF 3RD WORD.
     016130  010302                                                             MOV     R3,R2
     016132  062702  000010                                                     ADD     #10,R2
4092 016136                             WHILE (R3) GT #999. DO
     016136                                                                     50304$:
     016136  021327  001747                                                     CMP     (R3),#999.
     016142  003404                                                             BLE     50305$
4093 016144                               LET (R3) := (R3) - #1000.      ;UPDATE BYTE COUNT
     016144  162713  001750                                                     SUB     #1000.,(R3)
4094 016150                               LET (R2) := (R2) + #1          ;3RD WORD.
     016150  005212                                                             INC     (R2)
```

```
016152                        ENDDO
016152  000771                                                                BR      50304$
016154                                                                        50305$:
016154                        LET R3 := R2 + #10              ;R3 = ADR OF 4TH WORD.
016154  010203                                                                MOV     R2,R3
016156  062703  000010                                                        ADD     #10,R3
016162                        WHILE (R2) GT #999. DO
016162                                                                        50306$:
016162  021227  001747                                                        CMP     (R2),#999.
016166  003404                                                                BLE     50307$
016170                          LET (R2) := (R2) - #1000.     ;UPDATE BYTE COUNT
016170  162712  001750                                                        SUB     #1000.,(R2)
016174                          LET (R3) := (R3) + #1         ;4TH WORD.
016174  005213                                                                INC     (R3)
016176                        ENDDO
016176  000771                                                                BR      50306$
016200                                                                        50307$:
016200                        IFB RWERR NE #0 THEN            ;IF R/W ERROR, UPDATE ERROR COUNT.
016200  105737  003503                                                        TSTB    RWERR
016204  001440                                                                BEQ     50310$
016206                          LET R2 := R5 + EINC(R4) + #WRREC  ;R2 = ADR OF COUNTER.
016206  010502                                                                MOV     R5,R2
016210  066402  016316                                                        ADD     EINC(R4),R2
016214  062702  002720                                                        ADD     #WRREC,R2
016220                          IFB UNREC NE #0 THEN          ;IS THE ERROR UNRECOVERABLE?
016220  105737  003504                                                        TSTB    UNREC
016224  001404                                                                BEQ     50311$
016226                            LET R2 := R2 + #10          ;YES, POINT TO NEXT COUNTER.
016226  062702  000010                                                        ADD     #10,R2
016232                            LET (R2) := (R2) + #1       ;UPDATE THE ERROR COUNTER
016232  005212                                                                INC     (R2)
016234                          ELSE                          ;ELSE - IF ERROR IS RECOVERABLE:
016234  000424                                                                BR      50312$
016236                                                                        50311$:
016236                            LET (R2) := (R2) + #1       ;UPDATE THE ERROR COUNTER
016236  005212                                                                INC     (R2)
016240                            IFB IREC EQ #0 THEN         ;IF ERROR RECOVERY IS ENABLED:
016240  105737  002210                                                        TSTB    IREC
016244  001020                                                                BNE     50313$
016246                              IFB DROPED EQ #0 ANDB ERCVER NE #0 THEN ;IF UNIT HAS NOT BEEN DROPPED:
016246  105737  003541                                                        TSTB    DROPED
016252  001015                                                                BNE     50314$
016254  105737  002207                                                        TSTB    ERCVER
016260  001412                                                                BEQ     50314$
016262                                PRINTB #NURTY1,RETRYC   ;PRINT # OF RETRIES TO RECOVER
016262  013746  003474                                                                MOV     RETRYC,-(SP)
016266  012746  005167                                                                MOV     #NURTY1,-(SP)
016272  012746  000002                                                                MOV     #2,-(SP)
016276  010600                                                                        MOV     SP,R0
016300  104414                                                                        TRAP    C$PNTB
016302  062706  000006                                                                ADD     #6,SP
016306                              ENDIF                     ;PROVIDED PRINT HAS BEEN ENABLED
016306                                                                        50314$:
016306                            ENDIF
016306                                                                        50313$:
016306                          ENDIF
016306                                                                        50312$:
```

J9


```
 4114 016306                                ENDIF
      016306                                                                  50310$:
 4115 016306                                ENDIF
      016306                                                                  50300$:
 4116 016306                                ENDIF
      016306                                                                  50277$:
 4117 016306  000207                        RTS PC
 4118                              ;        INDEXES TO BYTE COUNTERS.
 4119 016310  000000               BINC:    0                                ;WRITE.
 4120 016312  000040                        40                               ;READ REV.
 4121 016314  000100                        100                              ;READ FWD.
 4122                              ;        INDEXES TO READ/WRITE ERROR COUNTERS.
 4123 016316  000000               EINC:    0                                ;WRITE.
 4124 016320  000020                        20                               ;READ REV.
 4125 016322  000040                        40                               ;READ FWD.
 4126
 4127
 4128
```

```
                                .SBTTL RAMER - READ AND DISPLAY SELECTED RAM
                        ;+
                        ;ROUTINE TO READ THE SELECTED RAM LOCATIONS
                        ;-
016324 010546           RAMER:: MOV     R5,-(SP)
016326 010446                   MOV     R4,-(SP)
016330 010346                   MOV     R3,-(SP)
016332 010246                   MOV     R2,-(SP)
016334 010146                   MOV     R1,-(SP)
016336 012701 003346            MOV     #RAMDATA,R1     ;ADDRESS TO SAVE THE RAM DATA
016342 013702 003342            MOV     RAMHLD,R2       ;BYTE ADDRESS OF THE FIRST RAM DATA
016346 013703 003406            MOV     RAMSIZ,R3       ;SET THE SIZE OF THE READ UP
016352 016504 002456            MOV     TSDB(R5),R4     ;MOV THE TSDB ADDRESS INTO R4
016356 005204                   INC     R4              ;ADD 1 TO IT
016360 000240           10$:    NOP
016362 004737 012124            JSR     PC,WSSR         ;WAIT FOR THE SSR TO SET
016366 110214                   MOVB    R2,(R4)         ;SELECT NEXT RAM ADDRESS
016370 004737 012124            JSR     PC,WSSR         ;WAIT FOR SSR TO SET
016374 117521 002456            MOVB    @TSBA(R5),(R1)+ ;READ THE RAM DATA
016400 062702 000001    20$:    ADD     #1,R2           ;ADDRESS OF THE NEXT RAM LOCATION
016404 077313                   SOB     R3,10$          ;NUMBER OF LOCATIONS COUNTER
016406 013704 003406            MOV     RAMSIZ,R4       ;GET THE RAM SIZE
016412 013702 003342            MOV     RAMHLD,R2       ;GET THE STARTING RAM ADDRESS
016416 060204                   ADD     R2,R4           ;CALCULATE THE END ADDRESS
016420 162704 000001            SUB     #1,R4           ;CORRECT VALUE OF PRINTOUT
016424                          PRINTX  #RAMIOP,R2,R4   ;RAM ADDRESS = 10 - 17, ETC.
016424 010446                                                           MOV     R4,-(SP)
016426 010246                                                           MOV     R2,-(SP)
016430 012746 005377                                                    MOV     #RAMIOP,-(SP)
016434 012746 000003                                                    MOV     #3,-(SP)
016440 010600                                                           MOV     SP,R0
016442 104415                                                           TRAP    C$PNTX
016444 062706 000010                                                    ADD     #10,SP
016450 012701 003346            MOV     #RAMDATA,R1     ;ADDRESS OF WHERE RAM DATA IS
016454 013703 003406            MOV     RAMSIZ,R3       ;THE SIZE OF THE RAM FIELD READ
016460 005004           30$:    CLR     R4              ;NO EXTRA DATA LEFT OVER
016462 112104                   MOVB    (R1)+,R4        ;PICK UP BYTE OF RAM DATA
016464 042704 177400            BIC     #177400,R4      ;GET RID OF SIGN EXTEND
016470                          PRINTX  #RAMPD,R4       ;"010 211 111 222 377 000 123 134 ETC."
016470 010446                                                           MOV     R4,-(SP)
016472 012746 005450                                                    MOV     #RAMPD,-(SP)
016476 012746 000002                                                    MOV     #2,-(SP)
016502 010600                                                           MOV     SP,R0
016504 104415                                                           TRAP    C$PNTX
016506 062706 000006                                                    ADD     #6,SP
016512 077316                   SOB     R3,30$          ;LOOP UNTIL ALL PRINTED
016514 012601                   MOV     (SP)+,R1
016516 012602                   MOV     (SP)+,R2
016520 012603                   MOV     (SP)+,R3
016522 012604                   MOV     (SP)+,R4
016524 012605                   MOV     (SP)+,R5
016526 000207           50$:    RTS     PC              ;RETURN
                        ;       IF A WRITE/VERIFY COMMAND IS ISSUED, CONTROL IS THEN
                        ;       TRANSFERRED TO THIS SUBROUTINE TO READ REVERSE, CHECK DATA,
                        ;       READ FORWARD, CHECK DATA, THEN CONTINUE TO NEXT COMMAND.
                        ;       INPUTS:
                        ;       OUTPUTS:
```

```
4174                                    ;       REGISTERS:
4175                                    ;       CALLS:          VFEXC.
4176
4177 016530                             VFYDAT::  LET CMPDAT := #0              ;BTL
     016530  005037  003410                                                             CLR     CMPDAT
4178 016534                                       IFB VFYFLG NE #0 THEN         ;IF DATA IS TO BE VERIFIED:
     016534  105737  003534                                                             TSTB    VFYFLG
     016540  001437                                                                     BEQ     50315$
4179 016542
     016542  105737  003544                                                             TSTB    STREAM
     016546  001015                                                                     BNE     50316$
4180 016550                                        LET PCMDWD := CMDWRD                 ;SAVE THE PREVIOUS COMMAND WORD.
     016550  013737  003426  003432                                                     MOV     CMDWRD,PCMDWD
4181 016556                                        LET CMDWRD := #RDR           ;COMMAND IS READ REV.
     016556  012737  104401  003426                                                     MOV     #RDR,CMDWRD
4182 016564                                        LET CMDLG  := #4             ;SET UP CMD LOGGING INDEX.
     016564  012737  000004  003434                                                     MOV     #4,CMDLG
4183 016572  004737  016642                        JSR PC,VFEXC                 ;GO READ ALL THE RECORDS REV.
4184 016576                                        LET CMPDAT := #0             ;BTL
     016576  005037  003410                                                             CLR     CMPDAT
4185 016602                                       ENDIF
     016602                                                                     50316$:
4186 016602                                      IF DEVTBL(R5) NE #NINUSE THEN
     016602  026527  002536  177774                                                     CMP     DEVTBL(R5),#NINUSE
     016610  001413                                                                     BEQ     50317$
4187 016612                                        LET PCMDWD := CMDWRD         ;SAVE THE PREVIOUS COMMAND WORD.
     016612  013737  003426  003432                                                     MOV     CMDWRD,PCMDWD
4188 016620                                        LET CMDWRD := #RDF           ;COMMAND IS READ FWD.
     016620  012737  104001  003426                                                     MOV     #RDF,CMDWRD
4189 016626                                        LET CMDLG  := #6             ;SET UP CMD LOGGING INDEX.
     016626  012737  000006  003434                                                     MOV     #6,CMDLG
4190 016634  004737  016642                         JSR   PC,VFEXC              ;GO READ ALL RECORDS FWD.
4191 016640                                        ENDIF
     016640                                                                     50317$:
4192 016640                                       ENDIF
     016640                                                                     50315$:
4193 016640  000207                               RTS     PC                   ;RETURN.
```

```
                                ;       SUBROUTINE TO EXECUTE THE READ AND VERIFY, FORWARD OR REVERSE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2
                                ;       CALLS:          CMDAC, FIRSTU, VFISU, NEXTU, CKHAE.

016642                          VFEXC:: LET CMDPKT := CMDWRD CLR.BY #BRF.C  ;COMMAND PACKET = READ REV OR FWD.
016642  013737  003426  002314                                                    MOV     CMDWRD,CMDPKT
016650  042737  004000  002314                                                    BIC     #BRF.C,CMDPKT
016656                                  IFB SWBFLG NE #0 THEN           ;IF BYTES ARE TO BE SWAPPED:
016656  105737  003536                                                            TSTB    SWBFLG
016662  001403                                                                  BEQ     50320$
016664                                    LET CMDPKT := CMDPKT SET.BY #SWB.C  ;SET SWAB BIT IN CMD PACKET.
016664  052737  010000  002314                                                    BIS     #SWB.C,CMDPKT
016672                                  ENDIF
016672                                                                          50320$:
016672                                  LET CMDSAV := CMDPKT            ;SAVE COMMAND PACKET 1ST WORD.
016672  013737  002314  003430                                                    MOV     CMDPKT,CMDSAV
016700  013737  003416  002316          MOV     DATARD,CMDPKT+CP.ADL    ;SAVE BUFFER START ADDRESS.
016706                                  LET NCNT := #0                  ;CLEAR NUMBER OF OPERATIONS.
016706  005037  003420                                                            CLR     NCNT
016712                                  WHILE NCNT LT NCNT1 DO          ;WHILE THERE ARE RECORDS REMAINING:
016712                                                                          50321$:
016712  023737  003420  003422                                                    CMP     NCNT,NCNT1
016720  002101                                                                  BGE     50322$
016722  004737  007672                    JSR PC,CMDAC                    ;STORE CMD ASCII IN ERROR MSG.
016726  105737  003544                    TSTB  STREAM                    ;CHECK IF WE ARE STREAMING
016732  001006                            BNE   1$                        ;BRANCH OVER DEVTBL CHECK. THIS ENABLES
                                                                ;US TO TEST ONE DRIVE AT A TIME.
016734  004737  017666                    JSR   PC,FIRSTU                 ;SET UP FOR FIRST UNIT.
016740                                    WHILE DEVTBL(R5) NE #END DO   ;WHILE THERE ARE DEVICES REMAINING:
016740                                                                          50323$:
016740  026527  002536  177777                                                    CMP     DEVTBL(R5),#END
016746  001445                                                                  BEQ     50324$
016750                          1$:             IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
016750  032737  000400  003426                                                    BIT     #MOD.CO,CMDWRD
016756  001421                                                                  BEQ     50325$
016760                                    IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
016760  032765  000002  003516                                                    BIT     #XO.BOT,EOTFLG(R5)
016766  001014                                                                  BNE     50326$
016770                                      IF #XO.EOT SETIN EOTFLG(R5) THEN   ;BUT IF AT EOT
016770  032765  000001  003516                                                    BIT     #XO.EOT,EOTFLG(R5)
016776  001406                                                                  BEQ     50327$
017000                                        IFB ALLEOT NE #0 THEN           ;AND ALL OTHERS AT EOT
017000  105737  003543                                                            TSTB    ALLEOT
017004  001402                                                                  BEQ     50330$
017006  004737  017126                        JSR PC,VFISU                    ;THEN READ VERIFY
017012                                      ENDIF                             ;IF NOT ALL AT EOT, FREEZE UNIT(S) AT E
017012                                                                          50330$:
017012                                    ELSE                              ;IF NOT AT BOT AND
017012  000402                                                                  BR      50331$
017014                                                                          50327$:
017014  004737  017126                      JSR PC,VFISU                      ;NOT AT EOT, READ VFY
017020                                    ENDIF
017020                                                                          50331$:
017020                                  ENDIF
017020                                                                          50326$:
```

```
4225 017020                                    ELSE                          ;ELSE IF CMD IS NOT REVERSE:
     017020  000412                                                                 BR      50332$
     017022                                                                 50325$:
4226 017022                                      IF @XO.EOT NOTSETIN EOTFLG(R5) OR @CMD.CO NOTSETIN CMDWRD THEN
     017022  032765  000001  003516                                                BIT     @XO.EOT,EOTFLG(R5)
     017030  001404                                                                BEQ     50333$
     017032  032737  000001  003426                                                BIT     @CMD.CO,CMDWRD
     017040  001002                                                                BNE     50334$
     017042                                                                 50333$:
4227
4228 017042  004737  017126                        JSR PC,VFISU                 ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
4229 017046                                      ENDIF                            ;ISSUE CMD, CHECK STATUS AND DATA.
     017046                                                                 50334$:
4230 017046                                    ENDIF
     017046                                                                 50332$:
4231 017046  105737  003544                    TSTB STREAM                  ;CHECK FOR TEST OF ON UNIT AT A TIME.
4232 017052  001003                            BNE 2$                       ;BRANCH, IF STREAMING TESTS.
4233 017054  004737  017734                    JSR PC,NEXTU                 ;GO FIND THE NEXT UNIT.
4234 017060                                    ENDDO
     017060  000727                                                                BR      50323$
     017062                                                                 50324$:
4235 017062  004737  020274          2$:       JSR   PC,CKHAE               ;CHECK FOR HALT AFTER EACH CMD.
4236 017066                                    IF DEVTBL(R5) EQ @NINUSE THEN ;IF DRIVES BEEN DROPPED EXIT
     017066  026527  002536  177774                                                CMP     DEVTBL(R5),@NINUSE
     017074  001005                                                                BNE     50335$
4237 017076                                    LET NCNT := NCNT1 - @1
     017076  013737  003422  003420                                                MOV     NCNT1,NCNT
     017104  005337  003420                                                        DEC     NCNT
4238 017110                                    ENDIF
     017110                                                                 50335$:
4239 017110                                    LET NCNT := NCNT + @1        ;UPDATE THE RECORD COUNT.
     017110  005237  003420                                                        INC     NCNT
4240 017114                                    LET PCMDWD := CMDWRD         ;SAVE PREVIOUS COMMAND WORD.
     017114  013737  003426  003432                                                MOV     CMDWRD,PCMDWD
4241
4242 017122                                  ENDDO
     017122  000673                                                                BR      50321$
     017124                                                                 50322$:
4243 017124  000207                          RTS     PC                     ;RETURN.
```

```
                                ;       SUBROUTINE TO ISSUE COMMAND, AWAIT INTERRUPT,
                                ;       CHECK STATUS, CHECK DATA.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2
                                ;       CALLS:          EXCUTE, GOWAIT, CKDATA.

017126                          VFISU::
017126                              LET R2 := DATARD + #8.      ;INIT READ BUFFER POINTER.
017126  013702  003416                                                  MOV     DATARD,R2
017132  062702  000010                                                  ADD     #8.,R2
017136                              WHILE R2 NE DATARD DO       ;UNTIL 8 BYTES HAVE BEEN SET,
017136                                                                  50336$:
017136  020237  003416                                                  CMP     R2,DATARD
017142  001403                                                          BEQ     50337$
017144                              LET -(R2) := #-1            ;INIT READ BUFFER.
017144  012742  177777                                                  MOV     #-1,-(R2)
017150                              ENDDO
017150  000772                                                          BR      50336$
017152                                                                  50337$:
017152  004737  010774              JSR PC,EXCUTE               ;GO EXECUTE THE COMMAND.
017156                              IFB DROPED EQ #0 THEN       ;IF UNIT HAS NOT BEEN DROPPED THEN:
017156  105737  003541                                                  TSTB    DROPED
017162  001002                                                          BNE     50340$
017164  004737  011470              JSR PC,GOWAIT               ;GO WAIT FOR DONE BIT.
017170                              ENDIF
017170                                                                  50340$:
017170                              IFB DROPED EQ #0 THEN       ;IF UNIT HAS NOT BEEN DROPPED THEN:
017170  105737  003541                                                  TSTB    DROPED
017174  001025                                                          BNE     50341$
017176                                IF #X0.BOT NOTSETIN EOTFLG(R5) THEN   ;WHEN NOT REVERSED INTO BOT, THEN
017176  032765  000002  003516                                          BIT     #X0.BOT,EOTFLG(R5)
017204  001021                                                          BNE     50342$
017206                                  IF L$TEST NE #3 THEN
017206  023727  002114  000003                                          CMP     L$TEST,#3
017214  001403                                                          BEQ     50343$
017216  004737  017252                    JSR   PC,CKDATA       ;GO VERIFY DATA.
017222                                  ELSE
017222  000412                                                          BR      50344$
017224                                                                  50343$:
017224  005237  003410                    INC   CMPDAT          ;ONE MORE XFER BEFORE A COMPARISON
017230                                    IF CMPDAT GT DATRAT THEN
017230  023737  003410  003412                                          CMP     CMPDAT,DATRAT
017236  003404                                                          BLE     50345$
017240  004737  017252                      JSR       PC,CKDATA
017244                                      LET CMPDAT := #0
017244  005037  003410                                                  CLR     CMPDAT
017250                                    ENDIF
017250                                                                  50345$:
017250                                  ENDIF
017250                                                                  50344$:
017250                                ENDIF
017250                                                                  50342$:
017250                              ENDIF
017250                                                                  50341$:
017250  000207                      RTS PC
```

```
4277                            ;       SUBROUTINE TO COMPARE DATA BETWEEN READ AND WRITE BUFFERS
4278                            ;       AND PRINT ERROR MESSAGE ON MISCOMPARE.
4279                            ;       INPUTS:
4280                            ;       OUTPUTS:
4281                            ;       REGISTERS:      R2, R3, R4.
4282                            ;       CALLS:          GCMDA
4283
4284 017252                    CKDATA:: LET R3 := BRFCNT - MSGPKT+MS.RFC ;COMPUTE REC LENGTH READ
      017252  013703  003424                                                MOV     BRFCNT,R3
      017256  163703  002344                                                SUB     MSGPKT+MS.RFC,R3
4285 017262                      IF R3 EQ #0 THEN                ;WHEN NO DATA RECEIVED
      017262  005703                                                        TST     R3
      017264  001015                                                        BNE     50346$
4286 017266                        ERRHRD 17,WTVERM,DTAERM               ;PRINT ERROR AND EXIT
      017266  104456                                                       TRAP    C$ERHRD
      017270  000021                                                       .WORD   17
      017272  004260                                                       .WORD   WTVERM
      017274  005472                                                       .WORD   DTAERM
4287 017276                        PRINTB #DTAER4                  ;COMPARE ROUTINE
      017276  012746  005104                                               MOV     #DTAER4,-(SP)
      017302  012746  000001                                               MOV     #1,-(SP)
      017306  010600                                                        MOV     SP,R0
      017310  104414                                                        TRAP    C$PNTB
      017312  062706  000004                                               ADD     #4,SP
4288 017316                      ELSE
      017316  000560                                                        BR      50347$
      017320                                                                50346$:
4289 017320                        IF R3 HI BRFCNT THEN            ;WHEN REC READ IS LONGER
      017320  020337  003424                                                CMP     R3,BRFCNT
      017324  101417                                                       BLOS    50350$
4290 017326                          ERRHRD 17,WTVERM,DTAERM             ;THAN EXPECTED, PRINT
      017326  104456                                                       TRAP    C$ERHRD
      017330  000021                                                       .WORD   17
      017332  004260                                                       .WORD   WTVERM
      017334  005472                                                       .WORD   DTAERM
4291 017336                          PRINTB #DTAER5,CMDPKT+CP.CNT      ;AN ERROR MESSAGE
      017336  013746  002322                                               MOV     CMDPKT+CP.CNT,-(SP)
      017342  012746  005125                                               MOV     #DTAER5,-(SP)
      017346  012746  000002                                               MOV     #2,-(SP)
      017352  010600                                                        MOV     SP,R0
      017354  104414                                                        TRAP    C$PNTB
      017356  062706  000006                                               ADD     #6,SP
4292 017362                      ELSE                            ;AND EXIT ROUTINE
      017362  000536                                                        BR      50351$
      017364                                                                50350$:
4293 017364                        LET CKDCNT := R3 - #1                 ;SAVE VERIFICATION LENGTH - 1.
      017364  010337  017662                                               MOV     R3,CKDCNT
      017370  005337  017662                                               DEC     CKDCNT
4294 017374  005037  017664       CLR CKDFF                       ;CLEAR # OF BYTES IN ERROR COUNTER.
4295 017400  005002               CLR R2                          ;INIT BYTE COUNTER
4296 017402                        LET R3 := DATAWT                ;GET WRITE BUFFER ADDRESS.
      017402  013703  003414                                                MOV     DATAWT,R3
4297 017406                        LET R4 := DATARD                ;GET READ BUFFER ADDRESS.
      017406  013704  003416                                                MOV     DATARD,R4
4298 017412                        IFB T1SWB NE #0 THEN            ;WHEN RUNNING TEST 1-SUB 12,
      017412  105737  003542                                                TSTB    T1SWB
      017416  001401                                                        BEQ     50352$
```

```
 4299 017420  000313                    SWAB (R3)                    ;SWAP FIRST WORD OF WRT BFR
 4300 017422                            ENDIF                        ;WHICH CONTAINS THE RECORD COUNT
      017422                                                                50352$:
 4301 017422                            REPEAT                       ;REPEAT UNTIL ALL DATA IS COMPARED:
      017422                                                                50353$:
 4302 017422                              IF R2 EQ CKDCNT THEN       ;IF THIS IS THE LAST BYTE THEN:
      017422  020237  017662                                                CMP     R2,CKDCNT
      017426  001011                                                       BNE     50354$
 4303 017430                                IFB SWBFLG NE #0 THEN    ;IF BYTE SWAPPING IS ENABLED THEN:
      017430  105737  003536                                                TSTB    SWBFLG
      017434  001406                                                       BEQ     50355$
 4304 017436                                  IF #BIT00 NOTSETIN CKDCNT THEN        ;IF RECORD LENGTH IS ODD THEN:
      017436  032737  000001  017662                                        BIT     #BIT00,CKDCNT
      017444  001002                                                       BNE     50356$
 4305 017446  105723                            TSTB        (R3)+             ;LAST BYTE WILL BE IN
 4306 017450  105724                            TSTB        (R4)+             ;THE UPPER BYTE.
 4307 017452                                  ENDIF
      017452                                                                50356$:
 4308 017452                                ENDIF
      017452                                                                50355$:
 4309 017452                              ENDIF
      017452                                                                50354$:
 4310 017452  121314                      CMPB (R3),(R4)               ;ARE THEY EQUAL.
 4311 017454  001452                      BEQ       3$                     ;BR IF SO.
 4312 017456  005737  017664              TST CKDFF                    ;1 ST TIME THRU?
 4313 017462  001010                      BNE       2$                     ;BR IF NOT.
 4314 017464  005265  003300              INC       VFYCNT(R5)             ;INC THE VERIFY ERROR COUNTER.
 4315 017470  005265  003310              INC HRDCNT(R5)               ;INC THE HARD ERROR COUNT.
 4316 017474                              ERRHRD 17,WTVERM,DTAERM      ;REPORT WRITE/VERIFY ERROR.
      017474  104456                                                          TRAP    C$ERHRD
      017476  000021                                                         .WORD   17
      017500  004260                                                         .WORD   WTVERM
      017502  005472                                                         .WORD   DTAERM
 4317 017504                      2$:     LET CKDFF := CKDFF + #1      ;INCREMENT # OF BYTES IN ERROR.
      017504  005237  017664                                                 INC     CKDFF
 4318 017510  111437  003444              MOVB (R4),TIME1              ;SAVE WAS DATA FOR TYPOUT.
 4319 017514  042737  177400  003444      BIC       #177400,TIME1          ;CLEAR GARBAGE.
 4320 017522  111337  003446              MOVB (R3),TIME2              ;SAVE SHOULD BE DATA FOR TYPOUT.
 4321 017526  042737  177400  003446      BIC       #177400,TIME2          ;CLEAR GARBAGE.
 4322 017534                              IF CKDFF LT #11. THEN        ;IF ERROR BYTE COUNT IS LESS THAN 11:
      017534  023727  017664  000013                                          CMP     CKDFF,#11.
      017542  002017                                                         BGE     50357$
 4323 017544                                PRINTX #DTAER2,R2,<B,TIME1>,<B,TIME2>  ;PRINT EXP + ACT DATA.
      017544  005046                                                         CLR     -(SP)
      017546  153716  003446                                                 BISB    TIME2,(SP)
      017552  005046                                                         CLR     -(SP)
      017554  153716  003444                                                 BISB    TIME1,(SP)
      017560  010246                                                         MOV     R2,-(SP)
      017562  012746  004773                                                 MOV     #DTAER2,-(SP)
      017566  012746  000004                                                 MOV     #4,-(SP)
      017572  010600                                                         MOV     SP,R0
      017574  104415                                                         TRAP    C$PNTX
      017576  062706  000012                                                 ADD     #12,SP
 4324 017602                              ENDIF
      017602                                                                50357$:
 4325 017602  105723              3$:     TSTB (R3)+                   ;UPDATE WRITE BUFFER ADDRESS.
 4326 017604  105724                      TSTB (R4)+                   ;UPDATE READ BUFFER ADDRESS.
```

```
4327 017606  105722                          TSTB (R2)+                          ;UPDATE BYTE COUNTER.
4328 017610                                  UNTIL R2 GT CKDCNT                  ;END OF DATA COMPARE REPEAT LOOP.
     017610  020237  017662                                                          CMP     R2,CKDCNT
     017614  003702                                                              BLE     50353$
4329 017616                                  LET CKDCNT := CKDCNT + #1           ;CKDCNT EQUALS RECORD LENGTH.
     017616  005237  017662                                                      INC     CKDCNT
4330 017622                                  IF CKDFF NE #0 THEN                 ;IF COMPARE ERROR HAS OCCURED THEN:
     017622  005737  017664                                                      TST     CKDFF
     017626  001414                                                              BEQ     50360$
4331 017630                                    PRINTB #DTAER3,CKDFF,CKDCNT       ;PRINT # OF BYTES IN ERROR.
     017630  013746  017662                                                      MOV     CKDCNT,-(SP)
     017634  013746  017664                                                      MOV     CKDFF,-(SP)
     017640  012746  005042                                                      MOV     #DTAER3,-(SP)
     017644  012746  000003                                                      MOV     #3,-(SP)
     017650  010600                                                              MOV     SP,R0
     017652  104414                                                              TRAP    C$PNTB
     017654  062706  000010                                                      ADD     #10,SP
4332 017660                                    ENDIF
     017660                                                                   50360$:
4333 017660                                  ENDIF
     017660                                                                   50351$:
4334 017660                                ENDIF
     017660                                                                   50347$:
4335 017660  000207                        RTS     PC                          ;OTHERWISE, RETURN.
4336
4337 017662  000000                CKDCNT: .WORD   0                           ;# OF BYTES TO BE VERIFIED -1.
4338 017664  000000                CKDFF:  .WORD   0                           ;# OF BYTES IN ERROR COUNTER.
```

```
4340                                    ;       SUBROUTINE TO FIND THE FIRST DEVICE IN THE TEST SEQUENCE.
4341                                    ;       INPUTS:
4342                                    ;       OUTPUTS:
4343                                    ;       REGISTERS:
4344                                    ;       CALLS:
4345
4346 017666                             FIRSTU::LET DROPED :B= #0               ;CLR UNIT DROPPED FLAG
     017666  105037  003541                                                           CLRB    DROPED
4347 017672                                     LET R5 := #0                    ;CLR DEVICE POINTER.
     017672  005005                                                                 CLR     R5
4348 017674                                     WHILE DEVTBL(R5) EQ #NINUSE DO  ;WHILE DEVICES ARE NOT IN USE:
     017674                                                                     50361$:
     017674  026527  002536  177774                                              CMP     DEVTBL(R5),#NINUSE
     017702  001003                                                             BNE     50362$
4349 017704                                       LET R5 := R5 + #2             ;POINT TO NEXT DEVICE.
     017704  062705  000002                                                       ADD     #2,R5
4350 017710                                     ENDDO
     017710  000771                                                             BR      50361$
     017712                                                                     50362$:
4351 017712                                     IF DEVTBL(R5) EQ #END THEN      ;IF ALL UNITS HAVE BEEN DROPPED THEN:
     017712  026527  002536  177777                                              CMP     DEVTBL(R5),#END
     017720  001001                                                             BNE     50363$
4352 017722                                       DOCLN                           ;DO CLEAN CODE AND TERMINATE PASS.
     017722  104444                                                               TRAP    C$DCLN
4353 017724                                     ENDIF
     017724                                                                     50363$:
4354 017724                                     LET L$LUN := DEVTBL(R5)         ;SET UNIT # IN "HEADER" FOR ERROR REPORT
     017724  016537  002536  002074                                              MOV     DEVTBL(R5),L$LUN
4355 017732  000207                            RTS     PC                      ;RETURN WITH 1ST DEVICE IN R5.
4356
4357
4358
4359
4360
4361                                    ;       SUBROUTINE TO FIND THE NEXT UNIT IN THE TEST CYCLE.
4362                                    ;       INPUTS:
4363                                    ;       OUTPUTS:
4364                                    ;       REGISTERS:
4365                                    ;       CALLS:
4366
4367 017734                             NEXTU:: LET DROPED :B= #0               ;CLR UNIT DROPPED FLAG
     017734  105037  003541                                                           CLRB    DROPED
4368 017740  042705  177770                     BIC     #177770,R5
4369 017744                                     REPEAT                          ;REPEAT UNTIL THE NEXT DEVICE IS FOUND.
     017744                                                                     50364$:
4370 017744                                       LET R5 := R5 + #2             ;UPDATE DEVICE TABLE POINTER.
     017744  062705  000002                                                       ADD     #2,R5
4371 017750                                     UNTIL DEVTBL(R5) NE #NINUSE
     017750  026527  002536  177774                                              CMP     DEVTBL(R5),#NINUSE
     017756  001772                                                             BEQ     50364$
4372 017760                                     LET L$LUN := DEVTBL(R5)         ;SET UNIT # IN "HEADER" FOR ERROR REPORT
     017760  016537  002536  002074                                              MOV     DEVTBL(R5),L$LUN
4373 017766  000207                            RTS     PC                      ;RETURN.
4374
4375
4376
```

```
                                ;       SUBROUTINE TO DROP A DEVICE FROM THE TEST SEQUENCE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:          MOVMSG, PRXST, LOG

4384 017770                     DROPU:: LET R5SAVE := R5                                        MOV     R5,R5SAVE
     017770  010537  003460
4385 017774                             LET FTLCNT(R5) := FTLCNT(R5) + #1  ;INCREMENT THE FATAL ERROR COUNT.
     017774  005265  003320                                                                     INC     FTLCNT(R5)
4386 020000                             LET R4 := MSGPKT+MS.XS3 CLR.BY #377 ;GET UDIAG ERROR CODE FROM XSTAT3.
     020000  013704  002354                                                                    MOV     MSGPKT+MS.XS3,R4
     020004  042704  000377                                                                    BIC     #377,R4
4387 020010                             LET R3 := MSGPKA(R5)            ;ADR OF THIS UNIT'S MSG PACKET.
     020010  016503  002506                                                                    MOV     MSGPKA(R5),R3
4388 020014                             LET R2 := #0                    ;CLR COUNTER.
     020014  005002                                                                            CLR     R2
4389 020016                             WHILE R2 NE #MSGCNT DO          ;WHILE THERE ARE MORE LOCATIONS:
     020016                                                                                    50365$:
     020016  020227  000016                                                                    CMP     R2,#MSGCNT
     020022  001405                                                                           BEQ     50366$
4390 020024                               LET (R3)+ := #-1              ;INIT THE MSG PACKET WITH ALL 1'S
     020024  012723  177777                                                                    MOV     #-1,(R3)+
4391 020030                               LET R2 := R2 + #2             ;UPDATE COUNTER.
     020030  062702  000002                                                                    ADD     #2,R2
4392 020034                             ENDDO
     020034  000770                                                                           BR      50365$
     020036                                                                                    50366$:
4393 020036                             LET @TSDB(R5) := #GSCPK         ;INITIATE A GET STATUS COMMAND.
     020036  012775  002324  002456                                                           MOV     #GSCPK,@TSDB(R5)
4394 020044  004737  012124             JSR PC,WSSR                    ;WAIT A WHILE FOR SSR=1
4395 020050  004737  012224             JSR PC,MOVMSG                  ;MOVE MSG PACKET TO COMMON AREA.
4396 020054                             IF R4 EQ #X3.RNY THEN          ;IF WE HAVE A CAPSTAN RUNAWAY THEN:
     020054  020427  157400                                                                    CMP     R4,#X3.RNY
     020060  001005                                                                           BNE     50367$
4397 020062                               ERRDF 16,RNYM,STAERM           ;REPORT CAPSTAN RUNAWAY WITH TACH CNT.
     020062  104455                                                                           TRAP    C$ERDF
     020064  000020                                                                           .WORD   16
     020066  004600                                                                           .WORD   RNYM
     020070  005640                                                                           .WORD   STAERM
4398 020072                             ELSE                           ;ELSE-IF NOT A RUNNAWAY:
     020072  000402                                                                           BR      50370$
     020074                                                                                    50367$:
4399 020074  004737  020212               JSR PC,PRXST                ;PRINT EXTENDED STATUS REGISTERS.
4400 020100                             ENDIF
     020100                                                                                    50370$:
4401 020100                             IFB RECLOG NE #0 THEN          ;IF THE RECORD HAS BEEN LOGGED THEN:
     020100  105737  003501                                                                   TSTB    RECLOG
     020104  001404                                                                           BEQ     50371$
4402 020106                               LET DROPED :B= DROPED + #1   ;SET UNIT DROPPED FLAG.
     020106  105237  003541                                                                   INCB    DROPED
4403 020112  004737  016024               JSR PC,LOG                   ;LOG DATA BYTES + RD/WR ERRORS.
4404 020116                             ENDIF
     020116                                                                                    50371$:
4405 020116                             DORPT                          ;PRINT PERFORMANCE REPORT
     020116  104424                                                                           TRAP    C$DRPT
4406 020120                     DROPUA: IF PASCNT(R5) NE #0 THEN
```

```
      020120  005765  003260                                                    TST     PASCNT(R5)
      020124  001402                                                            BEQ     50372$
 4407 020126                          LET PASCNT(R5) := PASCNT(R5) - #1
      020126  005365  003260                                                    DEC     PASCNT(R5)
 4408 020132                        ENDIF
      020132                                                                    50372$:
 4409 020132                        LET DROPN := DEVTBL(R5)          ;SAVE # OF UNIT TO BE DROPPED.
      020132  016537  002536  020210                                            MOV     DEVTBL(R5),DROPN
 4410 020140                        LET R0 := R5 SHIFT -1            ;R0=LOGICAL DEVICE NUMBER
      020140  010500                                                            MOV     R5,R0
      020142  006200                                                            ASR     R0
 4411 020144                        DODU R0                  ;DROP THE UNIT: EXEC BGNDU-ENDDU CODE IF IDU = 0
      020144  104451                                                            TRAP    C$DODU
 4412 020146                        IF DEVTBL(R5) NE #NINUSE THEN    ;IF UNIT NOT DROPPED
      020146  026527  002536  177774                                            CMP     DEVTBL(R5),#NINUSE
      020154  001410                                                            BEQ     50373$
 4413 020156                          IFB IREC EQ #0 THEN              ;IF RECOVERY IS ENABLED THEN:
      020156  105737  002210                                                    TSTB    IREC
      020162  001005                                                            BNE     50374$
 4414 020164  000240                    NOP
 4415 020166  000240                    NOP
 4416 020170  000240                    NOP
 4417 020172                            LET STAFLG :B= STAFLG + #1      ;SET START FLAG TO ENABLE REWIND.
      020172  105237  003546                                                    INCB    STAFLG
 4418 020176                          ENDIF
      020176                                                                    50374$:
 4419 020176                        ENDIF
      020176                                                                    50373$:
 4420 020176                        DRORTN: LET DROPED :B= DROPED + #1       ;SET UNIT DROPPED FLAG.
      020176  105237  003541                                                    INCB    DROPED
 4421 020202                                LET R5 := R5SAVE
      020202  013705  003460                                                    MOV     R5SAVE,R5
 4422 020206  000207                        RTS     PC                    ;RETURN.
 4423
 4424 020210  000000                DROPN:  .WORD   0                     ;# OF UNIT TO BE DROPPED
```

```
                                ;       SUBROUTINE TO PRINT EXTENDED STATUS REGISTERS.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

020212                          PRXST:: PRINTX #GETSTM
020212  012746  005253                                                  MOV     #GETSTM,-(SP)
020216  012746  000001                                                  MOV     #1,-(SP)
020222  010600                                                          MOV     SP,R0
020224  104415                                                          TRAP    C$PNTX
020226  062706  000004                                                  ADD     #4,SP
020232                                  PRINTX  #STAER5,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3
020232  013746  002354                                                  MOV     MSGPKT+MS.XS3,-(SP)
020236  013746  002352                                                  MOV     MSGPKT+MS.XS2,-(SP)
020242  013746  002350                                                  MOV     MSGPKT+MS.XS1,-(SP)
020246  013746  002346                                                  MOV     MSGPKT+MS.XS0,-(SP)
020252  012746  006465                                                  MOV     #STAER5,-(SP)
020256  012746  000005                                                  MOV     #5,-(SP)
020262  010600                                                          MOV     SP,R0
020264  104415                                                          TRAP    C$PNTX
020266  062706  000014                                                  ADD     #14,SP
020272  000207                                  RTS PC


                                ;       SUBROUTINE TO HALT AFTER EACH COMMAND.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R3, R4
                                ;       CALLS:

020274                          CKHAE:: IFB HAE NE #0 THEN              ;IF HALT FLAG IS SET:
020274  105737  002206                                                          TSTB    HAE
020300  001430                                                              BEQ     50375$
020302                                      IFB MISCFG EQ #0 THEN       ;
020302  105737  003551                                                          TSTB    MISCFG
020306  001023                                                              BNE     50376$
020310                                        MANUAL                    ;IS MANUAL INTERVENTION ALLOWED?
020310  104450                                                                  TRAP    C$MANI
020312                                        BNCOMPLETE CKHRTN           ;BR IF NOT.
020312  103023                                                                  BCC     CKHRTN
020314                                        LET R4 := CMDWRD            ;COMMAND WORD.
020314  013704  003426                                                          MOV     CMDWRD,R4
020320  004737  007746                        JSR PC,GCMDA                ;FETCH ADR OF CMD ASCII.
020324                                        LET HALTM :B= (R3)+         ;MOVE CMD ASCII
020324  112337  004136                                                          MOVB    (R3)+,HALTM
020330                                        LET HALTM+1 :B= (R3)+
020330  112337  004137                                                          MOVB    (R3)+,HALTM+1
020334                                        LET HALTM+2 :B= (R3)           ;INTO MESSAGE.
020334  111337  004140                                                          MOVB    (R3),HALTM+2
020340                                        GMANIL HALTM,TIME1,1,YES    ;HALT - WAIT FOR AN OEPRATOR INPUT.
020340  104443                                                                  TRAP    C$GMAN
020342  000404                                                                  BR      10000$
020344  003444                                                                  .WORD   TIME1
020346  000130                                                                  .WORD   T$CODE
```

```
GLOBAL AREAS    MACRO M1200  28-MAR-85 15:28  PAGE 92-1
RAMER - READ AND DISPLAY SELECTED RAM                                                    SEQ 0126

        020350  004136                                                          .WORD   HALTM
        020352  000001                                                          .WORD   1
        020354                          10000$:
   4455 020354                                  ELSE
        020354  000402                                                  BR      50377$
        020356                                                          50376$:
   4456 020356                                    LET MISCFG :B= #0     ;
        020356  105037  003551                                          CLRB    MISCFG
   4457 020362                                  ENDIF
        020362                                                          50377$:
   4458 020362                                ENDIF
        020362                                                          50375$:
   4459 020362  000207                  CKHRTN: RTS     PC                      ;RETURN
```

```
                                ;       SUBROUTINE TO CREATE THE SEQUENCE FOR A WRITE TAPE MARK
                                ;       COMMAND.  WILL EXECUTE COMMAND TO UUT.
                                ;       INPUTS:
                                ;       OUTPUTS: CMDSEQ
                                ;       CALLS:  SETUP, CMDAC, EXCUTE, GOWAIT

                                WRITEM::
                                        LET R1 := #CMDSEQ
020364  012701  003554                                                    MOV     #CMDSEQ,R1
                                        LET (R1)+ := #WTM       ;COMMAND
020370  012721  100011                                                    MOV     #WTM,(R1)+
                                        LET (R1)+ := #1         ;BRF
020374  012721  000001                                                    MOV     #1,(R1)+
                                        LET (R1)+ := #1         ;ITERATIONS
020400  012721  000001                                                    MOV     #1,(R1)+
                                        TST (R1)+               ;PATTERN
                                        LET (R1)+ := #END       ;TERMINATOR
020406  012721  177777                                                    MOV     #END,(R1)+
                                        LET R1 := #CMDSEQ       ;TOP OF BUFFER
020412  012701  003554                                                    MOV     #CMDSEQ,R1
020416  004737  010000                  JSR PC, SETUP           ;SET UP THE TABLE
020422  004737  007672                  JSR PC, CMDAC           ;LOAD THE ASCII
020426  004737  010774                  JSR PC, EXCUTE          ;ISSUE THE WTM COMMAND
020432  004737  011470                  JSR PC, GOWAIT          ;WAIT FOR THE COMMAND TO FINISH
020436  000207                          RTS PC                  ;RETURN TO CALLER

                                        .EVEN

                                        ENDMOD

020440
```

```
                                .TITLE MISCELLANEOUS SECTIONS
                                .SBTTL  REPORT CODING SECTION

020440                                  BGNMOD

                                ;++
                                ; THE REPORT CODING SECTION CONTAINS THE
                                ; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
                                ;--

020440                                  BGNRPT
020440                          L$RPT::

020440                                  LET     R5SAVE := R5            ;SAVE CURRENT DEVICE POINTER.
020440  010537  003460                                                          MOV     R5,R5SAVE
020444  004737  017666                  JSR     PC,FIRSTU               ;FIND THE FIRST UNIT.
020450                                  WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
020450                                                                          50400$:
020450  026527  002536  177777                                                  CMP     DEVTBL(R5),#END
020456  001562                                                                  BEQ     50401$
020460                                    PRINTS        #RPT1A,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
020460  016546  003330                                                          MOV     RECCNT(R5),-(SP)
020464  016546  003260                                                          MOV     PASCNT(R5),-(SP)
020470  016546  002536                                                          MOV     DEVTBL(R5),-(SP)
020474  012746  021302                                                          MOV     #RPT1A,-(SP)
020500  012746  000004                                                          MOV     #4,-(SP)
020504  010600                                                                  MOV     SP,R0
020506  104416                                                                  TRAP    C$PNTS
020510  062706  000012                                                          ADD     #12,SP
020514                          PRINTS        #RPT1B,WRBC+30(R5),WRBC+20(R5),WRBC+10(R5),WRBC(R5)
020514  016546  002560                                                          MOV     WRBC(R5),-(SP)
020520  016546  002570                                                          MOV     WRBC+10(R5),-(SP)
020524  016546  002600                                                          MOV     WRBC+20(R5),-(SP)
020530  016546  002610                                                          MOV     WRBC+30(R5),-(SP)
020534  012746  021357                                                          MOV     #RPT1B,-(SP)
020540  012746  000005                                                          MOV     #5,-(SP)
020544  010600                                                                  MOV     SP,R0
020546  104416                                                                  TRAP    C$PNTS
020550  062706  000014                                                          ADD     #14,SP
020554                          PRINTS        #RPT1C,RRBC+30(R5),RRBC+20(R5),RRBC+10(R5),RRBC(R5)
020554  016546  002620                                                          MOV     RRBC(R5),-(SP)
020560  016546  002630                                                          MOV     RRBC+10(R5),-(SP)
020564  016546  002640                                                          MOV     RRBC+20(R5),-(SP)
020570  016546  002650                                                          MOV     RRBC+30(R5),-(SP)
020574  012746  021430                                                          MOV     #RPT1C,-(SP)
020600  012746  000005                                                          MOV     #5,-(SP)
020604  010600                                                                  MOV     SP,R0
020606  104416                                                                  TRAP    C$PNTS
020610  062706  000014                                                          ADD     #14,SP
020614                          PRINTS        #RPT1D,RFBC+30(R5),RFBC+20(R5),RFBC+10(R5),RFBC(R5)
020614  016546  002660                                                          MOV     RFBC(R5),-(SP)
020620  016546  002670                                                          MOV     RFBC+10(R5),-(SP)
020624  016546  002700                                                          MOV     RFBC+20(R5),-(SP)
020630  016546  002710                                                          MOV     RFBC+30(R5),-(SP)
020634  012746  021501                                                          MOV     #RPT1D,-(SP)
020640  012746  000005                                                          MOV     #5,-(SP)
```

```
        020644  010600                                                          MOV     SP,R0
        020646  104416                                                          TRAP    C$PNTS
        020650  062706  000014                                                  ADD     #14,SP
4528    020654                  PRINTS      #RPT1F,WRREC(R5),RRREC(R5),RFREC(R5)
        020654  016546  002760                                                  MOV     RFREC(R5),-(SP)
        020660  016546  002740                                                  MOV     RRREC(R5),-(SP)
        020664  016546  002720                                                  MOV     WRREC(R5),-(SP)
        020670  012746  021605                                                  MOV     #RPT1F,-(SP)
        020674  012746  000004                                                  MOV     #4,-(SP)
        020700  010600                                                          MOV     SP,R0
        020702  104416                                                          TRAP    C$PNTS
        020704  062706  000012                                                  ADD     #12,SP
4529    020710                  PRINTS      #RPT1G,WRUNR(R5),RRUNR(R5),RFUNR(R5)
        020710  016546  002770                                                  MOV     RFUNR(R5),-(SP)
        020714  016546  002750                                                  MOV     RRUNR(R5),-(SP)
        02072C  016546  002730                                                  MOV     WRUNR(R5),-(SP)
        020724  012746  021656                                                  MOV     #RPT1G,-(SP)
        020730  012746  000004                                                  MOV     #4,-(SP)
        020734  010600                                                          MOV     SP,R0
        020736  104416                                                          TRAP    C$PNTS
        020740  062706  000012                                                  ADD     #12,SP
4530    020744                  IFB BADTSW NE #0 THEN       ;
        020744  105737  002211                                                  TSTB    BADTSW
        020750  001402                                                          BEQ     50402$
4531    020752  004737  021034    JSR PC,BTRPT        ;GO PRINT BAD TAPE SPOTS WHEN  ENABLED
4532    020756                  ENDIF
        020756                                                                  50402$:
4533    020756                  PRINTS      #RPT1I,SCCNT(R5),HRDCNT(R5),FTLCNT(R5),VFYCNT(R5)
        020756  016546  003300                                                  MOV     VFYCNT(R5),-(SP)
        020762  016546  003320                                                  MOV     FTLCNT(R5),-(SP)
        020766  016546  003310                                                  MOV     HRDCNT(R5),-(SP)
        020772  016546  003270                                                  MOV     SCCNT(R5),-(SP)
        020776  012746  022046                                                  MOV     #RPT1I,-(SP)
        021002  012746  000005                                                  MOV     #5,-(SP)
        021006  010600                                                          MOV     SP,R0
        021010  104416                                                          TRAP    C$PNTS
        021012  062706  000014                                                  ADD     #14,SP
4534    021016  004737  017734    JSR   PC,NEXTU              ;FIND THE NEXT UNIT.
4535    021022                  ENDDO
        021022  000612                                                          BR      50400$
        021024                                                                  50401$:
4536    021024                  LET R5 := R5SAVE            ;RESTORE CURRENT DEVICE POINTER.
        021024  013705  003460                                                  MOV     R5SAVE,R5
4537    021030                  EXIT    RPT
        021030  000167                                                          .WORD   J$JMP
        021032  001206                                                          .WORD   L10011-2-.
4538
4539                    ;       SUBR TO PRINT BAD TAPES SPOTS DURING THE REPORT PRINTS
4540                    ;       WRITE RETRIES: CUMULATIVE COUNT
4541                    ;       BAD TAPE SPOTS: COUNT PER TAPE PASS ONLY, NOT CUMULATIVE.
4542                    ;       COUNT OF RECOVERABLE WRITE ERRORS EXCLUDES BAD TAPE SPOTS.
4543
4544    021034          BTRPT:  PRINTS  #RPT1E,WRTYCT(R5)       ;PRINT GLOBAL WRITE RETRY COUNT
        021034  016546  003250                                                  MOV     WRTYCT(R5),-(SP)
        021040  012746  021727                                                  MOV     #RPT1E,-(SP)
        021044  012746  000002                                                  MOV     #2,-(SP)
        021050  010600                                                          MOV     SP,R0
```

```
     021052  104416                                                                         TRAP    C$PNTS
     021054  062706  000006                                                                 ADD     #6,SP
4545 021060                              LET BTPT := BTADDR(R5)  ;BTPT IS BOTH THE BAD TAPE SPOT COUNTER
     021060  016537  002550  003526                                                 MOV     BTADDR(R5),BTPT
4546 021066                              LET R3 := @BTPT SHIFT -1          ;AND THE LOGGING INDEX
     021066  017703  162434                                                         MOV     @BTPT,R3
     021072  006203                                                                 ASR     R3
4547 021074                              PRINTS  #RPT1J,R3         ;PRINT # OF BAD TAPE SPOTS
     021074  010346                                                                         MOV     R3,-(SP)
     021076  012746  021757                                                                 MOV     #RPT1J,-(SP)
     021102  012746  000002                                                                 MOV     #2,-(SP)
     021106  010600                                                                         MOV     SP,R0
     021110  104416                                                                         TRAP    C$PNTS
     021112  062706  000006                                                                 ADD     #6,SP
4548 021116                              IF R3 NE #0 THEN                  ;PRINT RECORD # IF BAD SPOTS DETECTED
     021116  005703                                                                 TST     R3
     021120  001457                                                                 BEQ     50403$
4549 021122                                IF R3 HI #20. THEN              ;
     021122  020327  000024                                                         CMP     R3,#20.
     021126  101402                                                                 BLOS    50404$
4550 021130                                  LET R3 := #20.                ;20 BAD SPOTS IS THE LIMIT
     021130  012703  000024                                                         MOV     #20.,R3
4551 021134                                ENDIF
     021134                                                                         50404$:
4552 021134                                PRINTS        #CRLFSP           ;
     021134  012746  005312                                                                 MOV     #CRLFSP,-(SP)
     021140  012746  000001                                                                 MOV     #1,-(SP)
     021144  010600                                                                         MOV     SP,R0
     021146  104416                                                                         TRAP    C$PNTS
     021150  062706  000004                                                                 ADD     #4,SP
4553 021154                                LET R4 := BTPT + #2   ;FETCH A BAD SPOT ID
     021154  013704  003526                                                         MOV     BTPT,R4
     021160  062704  000002                                                         ADD     #2,R4
4554 021164                                LET R2 := #0           ;R2 = PRINT COUNT PER LINE: 10 MAX
     021164  005002                                                                 CLR     R2
4555 021166                                REPEAT                          ;
     021166                                                                         50405$:
4556 021166                                  PRINTS      #RPT1K,(R4)     ;PRINT A BAD SPOT ID
     021166  011446                                                                         MOV     (R4),-(SP)
     021170  012746  022037                                                                 MOV     #RPT1K,-(SP)
     021174  012746  000002                                                                 MOV     #2,-(SP)
     021200  010600                                                                         MOV     SP,R0
     021202  104416                                                                         TRAP    C$PNTS
     021204  062706  000006                                                                 ADD     #6,SP
4557 021210                                  LET R2 := R2 + #1   ;COUNT PRINTS
     021210  005202                                                                 INC     R2
4558 021212                                  LET R4 := R4 + #2   ;NEXT
     021212  062704  000002                                                         ADD     #2,R4
4559 021216                                  IF R2 EQ #10. THEN  ;
     021216  020227  000012                                                         CMP     R2,#10.
     021222  001014                                                                 BNE     50406$
4560 021224                                    PRINTS #CRLFSP            ;GO TO NEXT PRINT LINE PAST 10 PRINTS
     021224  012746  005312                                                                 MOV     #CRLFSP,-(SP)
     021230  012746  000001                                                                 MOV     #1,-(SP)
     021234  010600                                                                         MOV     SP,R0
     021236  104416                                                                         TRAP    C$PNTS
     021240  062706  000004                                                                 ADD     #4,SP
```

```
4561 021244                                    LET R3 := R3 - #10.      ;ADJUST BAD SPOT COUNT
     021244  162703  000012                                                SUB     #10.,R3
4562 021250                                    LET R2 := R2 - #10.      ;ADJUST PRINT COUNT
     021250  162702  000012                                                SUB     #10.,R2
4563 021254                                  ENDIF                      ;
     021254                                                              50406$:
4564 021254                                UNTIL R2 EQ R3               ;LIMIT: # OF BAD SPOTS
     021254  020203                                                      CMP     R2,R3
     021256  001343                                                      BNE     50405$
4565 021260                                ENDIF                        ;
     021260                                                              50403$:
4566 021260                                PRINTS  #CRLF                ;
     021260  012746  005307                                                      MOV     #CRLF,-(SP)
     021264  012746  000001                                                      MOV     #1,-(SP)
     021270  010600                                                              MOV     SP,R0
     021272  104416                                                              TRAP    C$PNTS
     021274  062706  000004                                                      ADD     #4,SP
4567 021300  000207                        RTS PC
4568
4580                                       .NLIST  BEX
4581 021302     045     116     045  RPT1A: .ASCIZ  /%N%N%AUNIT %D1%S3%APASS:%D5%S3%ARECORD:%D5%N/
4582 021357     045     101     102  RPT1B: .ASCIZ  /%ABYTES WRITTEN  %D3%A,%Z3%A,%Z3%A,%Z3%N/
4583 021430     045     101     102  RPT1C: .ASCIZ  /%ABYTES READ REV %D3%A,%Z3%A,%Z3%A,%Z3%N/
4584 021501     045     101     102  RPT1D: .ASCII  /%ABYTES READ FWD %D3%A,%Z3%A,%Z3%A,%Z3%N/
4585 021551     045     123     062         .ASCIZ  /%S23%AWRT%S4%ARDR%S4%ARDF%N/
4586 021605     045     101     122  RPT1F: .ASCIZ  /%ARECOVERABLE ERRORS   %D5%S2%D5%S2%D5%N/
4587 021656     045     101     125  RPT1G: .ASCIZ  /%AUNRECOVERABLE ERRORS %D5%S2%D5%S2%D5%N/
4588 021727     045     101     127  RPT1E: .ASCIZ  /%AWRITE RETRIES%S8%D5%N/
4589 021757     045     116     045  RPT1J: .ASCIZ  /%N%D3%A BAD SPOTS THIS PASS PRECEDING RECORD #:/
4590 022037     045     104     065  RPT1K: .ASCIZ  /%D5%S1/
4591 022046     045     101     123  RPT1I: .ASCII  "%ASPEC COND%S3%AHARD%S3%AFATAL%S3%ACOMPARE%N"
4592 022122     045     123     063         .ASCIZ  /%S3%D5%S3%D5%S3%D5%S3%D5%N%N/
4593 022157     045     116     045  TAPCAP: .ASCIZ  /%N%ATAPE IN CARTRIDGE MUST BE 600' IN LENGTH.%N%N/
4594                                       .LIST   BEX
4595                                       .EVEN
4596 022242                                ENDRPT
     022242                          L10011:
     022242  104425                                                                  TRAP    C$RPT
4597
4598                                    ;LOAD DEVICE PROTECTION TABLE
4599                                    ;TABLE FOR SUPERVISOR TO IDENTIFY THE P-TBL FOR THE LOAD DEV
4600                                    ;THE SUPERVISOR USES THE TBL TO WARN THE OPERATOR WHEN HE TRIES TO TEST THE LOAD DEV
4601
4602 022244                                BGNPROT
     022244                          L$PROT::
4603 022244  000000                            .WORD 0              ;P-TBL OFFSET OF TSSR, THE TK25 CSR
4604 022246  177777                            .WORD -1             ;P-TBL OFFSET OF MASS BUS UNIT #: -1 = NOT A MASS BUS DEV
4605 022250  177777                            .WORD -1             ;P-TBL OFFSET OF DRIVE #: -1 = NONE, ONE DRIVE PER UNIBUS AD
4606 022252                                ENDPROT
```

```
                                .SBTTL  INITIALIZE SECTION

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

022252                                  BGNINIT
022252                          L$INIT::

                                INIT10: IF #BIT0!BIT1 SETIN #CMDPKT THEN  ;IF CMD PACKET IS NOT ON MODULO 4 BOUNDRY:
022252  032727  000003  002314                                                BIT     #BIT0!BIT1,#CMDPKT
022260  001421                                                                BEQ     50407$
                                  ERRSF 1,CMDPKM                ;PRINT ERROR MSG,
022262  104454                                                                TRAP    C$ERSF
022264  000001                                                                .WORD   1
022266  004176                                                                .WORD   CMDPKM
022270  000000                                                                .WORD   0
                                  DELAY 20.                     ;GO TO SUPERVISOR, WAIT 2 SECONDS.
022272  012727  000024                                                        MOV     #20.,(PC)+
022276  000000                                                                .WORD   0
022300  013727  002116                                                        MOV     L$DLY,(PC)+
022304  000000                                                                .WORD   0
022306  005367  177772                                                        DEC     -6(PC)
022312  001375                                                                BNE     .-4
022314  005367  177756                                                        DEC     -22(PC)
022320  001367                                                                BNE     .-20
                                  BR    INIT10                  ;
                                ENDIF
022324                                                                  50407$:

                                IFB CLRFLG NE #0 THEN           ;IF CLR COUNTERS FLAG SET:
022324  105737  002204                                                        TSTB    CLRFLG
022330  001413                                                                BEQ     50410$
                                  CLRB  CLRFLG                  ;INIT CLR FLAG.
                                  LET R2 := #0                                  CLR     R2

                                  WHILE R2 NE #CNTLEN DO                50411$:
022340                                                                  CMP     R2,#CNTLEN
022340  020227  000550                                                        BEQ     50412$
022344  001405
                                    LET WRBC(R2) := #0          ;CLR ALL STATISTICAL COUNTERS.
022346  005062  002560                                                        CLR     WRBC(R2)
                                    LET R2 := R2 + #2                           ADD     #2,R2

                                  ENDDO                                 BR      50411$
022356  000770                                                          50412$:
022360
                                ENDIF                                   50410$:

                                IFB RRANV NE #0 THEN            ;IF RESET RANDOM VARIABLE FLAG IS SET THEN:
022360  105737  002205                                                        TSTB    RRANV
022364  001406                                                                BEQ     50413$
                                  LET RANB := #RANBC              ;RESET RANDOM BASE #.
022366  012737  153624  003440                                                MOV     #RANBC,RANB
                                  LET RANS := #RANSC              ;RESET RANDOM SAVE LOCATION.
```

```
        022374  012737  032561  003442                                          MOV     #RANSC,RANS
4644    022402                          ENDIF
        022402                                                                  50413$:
4645    022402                          CLOCK   L,R5                    ;LINE CLOCK PRESENT?
        022402  012700  000114                                                  MOV     #'L,R0
        022406  104462                                                         TRAP    C$CLCK
        022410  010005                                                         MOV     R0,R5
4646    022412                          BNCOMPLETE      1$              ;SKIP NEXT IF NOT
        022412  103036                                                         BCC     1$
4647    022414                          LET LCSR := (R5)+
        022414  012537  003464                                                  MOV     (R5)+,LCSR
4648    022420                          LET LBRVEC := (R5)+
        022420  012537  003466                                                  MOV     (R5)+,LBRVEC
4649    022424                          LET LBRVEC := LBRVEC SHIFT 5
        022424  006337  003466                                                  ASL     LBRVEC
        022430  006337  003466                                                  ASL     LBRVEC
        022434  006337  003466                                                  ASL     LBRVEC
        022440  006337  003466                                                  ASL     LBRVEC
        022444  006337  003466                                                  ASL     LBRVEC
4650    022450                          LET LVECT := (R5)+
        022450  012537  003470                                                  MOV     (R5)+,LVECT
4651    022454                          LET LHERTZ := (R5)+
        022454  012537  003472                                                  MOV     (R5)+,LHERTZ
4652    022460                          SETVEC  LVECT,#CLKSVC,LBRVEC    ;SET UP FOR LINE CLOCK SERVICE
        022460  013746  003466                                                  MOV     LBRVEC,-(SP)
        022464  012746  010746                                                  MOV     #CLKSVC,-(SP)
        022470  013746  003470                                                  MOV     LVECT,-(SP)
        022474  012746  000003                                                  MOV     #3,-(SP)
        022500  104437                                                          TRAP    C$SVEC
        022502  062706  000010                                                  ADD     #10,SP
4653    022506  000402                  BR      2$
4654    022510                  1$:     LET LCSR := #0
        022510  005037  003464                                                  CLR     LCSR
4655    022514                  2$:
4656    022514                          READEF  #EF.START               ;READ START COMMAND EVENT FLAG.
        022514  012700  000040                                                  MOV     #EF.START,R0
        022520  104447                                                          TRAP    C$REFG
4657    022522                          BNCOMPLETE      INIT15          ;BRANCH IF NOT STARTING.
        022522  103030                                                          BCC     INIT15
4658    022524                          LET STAFLG :B= STAFLG + #1      ;SET START COMMAND FLAG.
        022524  105237  003546                                                  INCB    STAFLG
4659    022530                          LET HERE :B= #0                 ;RESET THE TEST 3 ASCII SEMAPHORE
        022530  105037  003340                                                  CLRB    HERE
4660    022534                          LET R5 := #6
        022534  012705  000006                                                  MOV     #6,R5
4661    022540                          REPEAT                          ;INITIATE UNIT NUMBER TABLE
        022540                                                                  50414$:
4662    022540                            LET DEVTBL(R5) := #NINUSE     ;BY STORING NOT IN USE IN EACH LOCATION.
        022540  012765  177774  002536                                          MOV     #NINUSE,DEVTBL(R5)
4663    022546                            LET R5 := R5 - #2
        022546  162705  000002                                                  SUB     #2,R5
4664    022552                          UNTIL R5 EQ #0
        022552  005705                                                          TST     R5
        022554  001371                                                          BNE     50414$
4665    022556                          LET R5 := L$UNIT SHIFT 1
        022556  013705  002012                                                  MOV     L$UNIT,R5
        022562  006305                                                          ASL     R5
```

```
4666 022564                      REPEAT                              ;STORE ALL UNIT
     022564                                                                   50415$:
4667 022564                        LET R5 := R5 - #2                 ;NUMBERS IN DEVTBL.
     022564 162705 000002                                                     SUB     #2,R5
4668 022570                        LET DEVTBL(R5) := R5 SHIFT -1
     022570 010565 002536                                                     MOV     R5,DEVTBL(R5)
     022574 006265 002536                                                     ASR     DEVTBL(R5)
4669 022600                      UNTIL R5 EQ #0
     022600 005705                                                            TST     R5
     022602 001370                                                            BNE     50415$
4670
4671 022604             INIT15: READEF #EF.PWR                       ;HAS THERE BE A POWER FAILURE?
     022604 012700 000034                                                     MOV     #EF.PWR,R0
     022610 104447                                                            TRAP    C$REFG
4672 022612                      BNCOMPLETE INIT16                   ;BRANCH IF NOT.
     022612 103004                                                            BCC     INIT16
4673 022614                        LET STAFLG :B= STAFLG + #1          ;IF SO - SET THE START FLAG.
     022614 105237 003546                                                     INCB    STAFLG
4674 022620                        LET PWRFLG :B= PWRFLG + #1        ;IF SO - SET THE POWER FAIL FLAG.
     022620 105237 003547                                                     INCB    PWRFLG
4675
4676 022624             INIT16: RFLAGS  OPFLAG                       ;READ AND STORE FLAGS SET BY OPERATOR
     022624 104421                                                            TRAP    C$RFLA
     022626 010037 003552                                                     MOV     R0,OPFLAG
4677 022632                     LET R3 := #0                         ;CLEAR EVENT FLAG
     022632 005003                                                            CLR     R3
4678 022634                         IFB PWRFLG EQ #0 THEN            ;IF POWER FAIL HAS NOT OCCURRED THEN:
     022634 105737 003547                                                     TSTB    PWRFLG
     022640 001020                                                            BNE     50416$
4679 022642                           READEF #EF.NEW                 ;UPDATE PASS COUNT WHEN
     022642 012700 000035                                                     MOV     #EF.NEW,R0
     022646 104447                                                            TRAP    C$REFG
4680 022650                           IFCOND CS THEN                 ;SUPERVISOR IS IN NEW PASS
     022650 103014                                                            BCC     50417$
4681 022652                             IFB STAFLG EQ #0 THEN        ;AND DIAG WAS NEITHER STARTED
     022652 105737 003546                                                     TSTB    STAFLG
     022656 001010                                                            BNE     50420$
4682 022660                               READEF #EF.RES             ;NOR
     022660 012700 000037                                                     MOV     #EF.RES,R0
     022664 104447                                                            TRAP    C$REFG
4683 022666                               IFCOND CC THEN             ;RESTARTED
     022666 103402                                                            BCS     50421$
4684 022670                                 LET R3 := COMP R3        ;DO IT
     022670 005103                                                            COM     R3
4685 022672                               ELSE
     022672 000401                                                            BR      50422$
     022674                                                                   50421$:
4686 022674                                 LET R3 := R3 + #1        ;SET 1ST PASS IF NEW PASS AND
     022674 005203                                                            INC     R3
4687 022676                               ENDIF                      ;RESTARTING
     022676                                                                   50422$:
4688 022676                             ELSE
     022676 000401                                                            BR      50423$
     022700                                                                   50420$:
4689 022700                               LET R3 := R3 + #1          ;SET 1ST PASS IF NEW PASS AND
     022700 005203                                                            INC     R3
4690 022702                             ENDIF                        ;STARTING
```

```
022702                                                                  50423$:
022702                                 ENDIF                      ;DO NOT UPDATE IT ON CONTINUE
022702                                                                  50417$:
022702                               ENDIF                        ;OR ON POWER FAIL
022702                                                                  50416$:
022702  004737  017666          JSR     PC,FIRSTU                 ;INIT DEVICE POINTER.
022706                          LET R2 := #0                      ;INIT DEVICE COUNTER.
022706  005002                                                          CLR     R2
022710                          WHILE DEVTBL(R5) NE #END DO
022710                                                                  50424$:
022710  026527  002536  177777                                          CMP     DEVTBL(R5),#END
022716  001450                                                          BEQ     50425$
022720                            LET R2 := R2 + #1
022720  005202                                                          INC     R2
022722                            LET R0 := R5 SHIFT -1
022722  010500                                                          MOV     R5,R0
022724  006200                                                          ASR     R0
022726                            GPHARD  R0,R0                   ;GET HARDWARE P TABLE FROM SUPER.
022726  104442                                                          TRAP    C$GPHRD
022730                            IFCOND CS THEN
022730  103036                                                          BCC     50426$
022732                              LET TSSR(R5) := (R0)          ;SAVE TSSR ADDRESS.
022732  011065  002466                                                  MOV     (R0),TSSR(R5)
022736                              LET TSDB(R5) := (R0)+ - #2    ;SAVE TSDB ADDRESS.
022736  012065  002456                                                  MOV     (R0)+,TSDB(R5)
022742  162765  000002  002456                                          SUB     #2,TSDB(R5)
022750                              LET TSVCT(R5) := (R0)         ;SAVE INTERRUPT VECTOR ADDRESS.
022750  011065  002476                                                  MOV     (R0),TSVCT(R5)
022754                              SETVEC TSVCT(R5),TS4INT(R5),#INTPRI ;SET UP INTERUPT PROCESSING CONDITIONS.
022754  012746  000340                                                          MOV     #INTPRI,-(SP)
022760  016546  002516                                                          MOV     TS4INT(R5),-(SP)
022764  016546  002476                                                          MOV     TSVCT(R5),-(SP)
022770  012746  000003                                                          MOV     #3,-(SP)
022774  104437                                                                  TRAP    C$SVEC
022776  062706  000010                                                          ADD     #10,SP
023002                              IF R3 NE #0 THEN              ;ACTUAL PASSCOUNT UPDATE PER R3
023002  005703                                                          TST     R3
023004  001410                                                          BEQ     50427$
023006                                IF R3 LT #0 THEN
023006  005703                                                          TST     R3
023010  002003                                                          BGE     50430$
023012                                  LET PASCNT(R5) := PASCNT(R5) + #1
023012  005265  003260                                                  INC     PASCNT(R5)
023016                                ELSE
023016  000403                                                          BR      50431$
023020                                                                  50430$:
023020                                  LET PASCNT(R5) := #1
023020  012765  000001  003260                                          MOV     #1,PASCNT(R5)
023026                                ENDIF
023026                                                                  50431$:
023026                              ENDIF
023026                                                                  50427$:
023026                            ENDIF
023026                                                                  50426$:
023026                            LET RECCNT(R5) := #0            ;CLEAR RECORD COUNT
023026  005065  003330                                                  CLR     RECCNT(R5)
023032  004737  017734          JSR   PC,NEXTU                    ;DO IT FOR ALL DEVICES.
```

```
4714 023036                             ENDDO
     023036  000724                                          BR      50424$
     023040                                                 50425$:
4715
4716 023040                             IF R2 EQ #0 THEN    ;IF THERE ARE NO UNITS:
     023040  005702                                          TST     R2
     023042  001026                                          BNE     50432$
4717 023044                               PRINTF #AUDRPM    ;PRINT ALL UNITS DROPPED,
     023044  012746  004741                                  MOV     #AUDRPM,-(SP)
     023050  012746  000001                                  MOV     #1,-(SP)
     023054  010600                                          MOV     SP,R0
     023056  104417                                          TRAP    C$PNTF
     023060  062706  000004                                  ADD     #4,SP
4718 023064                               DELAY 20.         ;GO TO SUPERVISOR, WAIT 2 SECONDS.
     023064  012727  000024                                  MOV     #20.,(PC)+
     023070  000000                                          .WORD   0
     023072  013727  002116                                  MOV     L$DLY,(PC)+
     023076  000000                                          .WORD   0
     023100  005367  177772                                  DEC     -6(PC)
     023104  001375                                          BNE     .-4
     023106  005367  177756                                  DEC     -22(PC)
     023112  001367                                          BNE     .-20
4719 023114                               BREAK             ;GO TO SUPERVISOR, CHECK TTY.
     023114  104422                                          TRAP    C$BRK
4720 023116                             DOCLN               ;DO CLEAN CODE + ABORT PASS.
     023116  104444                                          TRAP    C$DCLN
4721 023120                             ENDIF
     023120                                                 50432$:
4722
4723 023120                             SETPRI #PRI00               ;LOWER CPU PRIORITY TO 0
     023120  012700  000000                                          MOV     #PRI00,R0
     023124  104441                                                  TRAP    C$SPRI
4724 023126                             IFB IREC EQ #0 AND #ADR NOTSETIN OPFLAG THEN    ;IF ERROR RECOVERY IS ENABLED
     023126  105737  002210                                          TSTB    IREC
     023132  001145                                                  BNE     50433$
     023134  032737  000020  003552                                  BIT     #ADR,OPFLAG
     023142  001141                                                  BNE     50433$
4725 023144  004737  017666               JSR PC,FIRSTU              ;AND AUTO-DROP NOT CALLED, THEN SET UP FOR FIRST UNI
4726 023150                               WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
     023150                                                         50434$:
     023150  026527  002536  177777                                  CMP     DEVTBL(R5),#END
     023156  001533                                                  BEQ     50435$
4727 023160                                 BEGIN COUNTER           ;START 3.5 MINUTE COUNTER
4728 023160                                   INCR TIME1 FROM #1 TO #25 BY #1
     023160  012737  000001  003444                                  MOV     #1,TIME1
     023166  000402                                                  BR      50437$
     023170                                                         50440$:
     023170  005237  003444                                        INC     TIME1
     023174                                                         50437$:
     023174  023727  003444  000025                                CMP     TIME1,#25
     023202  003106                                                BGT     50441$
4729 023204                                   LET @TSDB(R5) := #GSCPK ;AND GET UNITS STATUS
     023204  012775  002324  002456                                MOV     #GSCPK,@TSDB(R5)
4730 023212                                   DELAY 250.            ;WAIT 25 MSEC.
     023212  012727  000372                                        MOV     #250.,(PC)+
     023216  000000                                                .WORD   0
     023220  013727  002116                                        MOV     L$DLY,(PC)+
```

```
        023224  000000
        023226  005367  177772                                                  .WORD   0
        023232  001375                                                          DEC     -6(PC)
        023234  005367  177756                                                  BNE     .-4
        023240  001367                                                          DEC     -22(PC)
                                                                                BNE     .-20
  4731  023242
        023242  032775  000200  002466    IF #TS.SSR SETIN @TSSR(R5) THEN
                                                                        BIT     #TS.SSR,@TSSR(R5)
        023250  001420                                                 BEQ     50442$
  4732  023252
        023252  032775  000100  002466      IF #TS.OFL NOTSETIN @TSSR(R5) THEN
                                                                        BIT     #TS.OFL,@TSSR(R5)
        023260  001001                                                 BNE     50443$
  4733  023262
        023262  000456                        LEAVE COUNTER           ;EXIT COUNTER WHEN UNIT ON LINE
                                                                        BR      50436$
  4734  023264                              ELSE
        023264                                                          50443$:
  4735  023264                                PRINTF #OFLINM,DEVTBL(R5) ;PRINT UNIT OFF LINE EVERY 10 SEC
        023264  016546  002536                                                  MOV     DEVTBL(R5),-(SP)
        023270  012746  005223                                                  MOV     #OFLINM,-(SP)
        023274  012746  000002                                                  MOV     #2,-(SP)
        023300  010600                                                          MOV     SP,R0
        023302  104417                                                          TRAP    C$PNTF
        023304  062706  000006                                                  ADD     #6,SP
  4736  023310                                ENDIF
        023310                                                          50444$:
  4737  023310                              ELSE
        023310  000412                                                  BR      50445$
        023312                                                          50442$:
  4738  023312                                PRINTF #NRDYM,DEVTBL(R5)
        023312  016546  002536                                                  MOV     DEVTBL(R5),-(SP)
        023316  012746  024244                                                  MOV     #NRDYM,-(SP)
        023322  012746  000002                                                  MOV     #2,-(SP)
        023326  010600                                                          MOV     SP,R0
        023330  104417                                                          TRAP    C$PNTF
        023332  062706  000006                                                  ADD     #6,SP
  4739  023336                              ENDIF
        023336                                                          50445$:
  4740  023336                            INCR TIME2 FROM #1 TO #13 BY #1
        023336  012737  000001  003446                                  MOV     #1,TIME2
        023344  000402                                                  BR      50446$
        023346                                                          50447$:
        023346  005237  003446                                         INC     TIME2
        023352                                                          50446$:
        023352  023727  003446  000013                                 CMP     TIME2,#13
        023360  003016                                                 BGT     50450$
  4741  023362                              DELAY 200.                ;WAIT FOR UNIT TO BE SET ON-LINE
        023362  012727  000310                                                  MOV     #200.,(PC)+
        023366  000000                                                          .WORD   0
        023370  013727  002116                                                  MOV     L$DLY,(PC)+
        023374  000000                                                          .WORD   0
        023376  005367  177772                                                  DEC     -6(PC)
        023402  001375                                                          BNE     .-4
        023404  005367  177756                                                  DEC     -22(PC)
        023410  001367                                                          BNE     .-20
  4742  023412                              BREAK                     ;ALLOW TERMINAL INTERRUPT
        023412  104422                                                          TRAP    C$BRK
  4743  023414                            ENDINC
        023414  000754                                                BR      50447$
```

```
      023416                                                          50450$:
 4744 023416                          ENDINC
      023416 000664                                                   BR      50440$
      023420                                                          50441$:
 4745 023420                        END COUNTER
      023420                                                          50436$:
 4746 023420                        IF TIME1 GT #25 THEN         ;IF OFF LINE FOR 3.5 MINUTES
      023420 023727 003444 000025                                     CMP     TIME1,#25
      023426 003404                                                   BLE     50451$
 4747 023430 004737 012224            JSR PC,MOVMSG              ;GET MESSAGE PACKET
 4748 023434 004737 013330            JSR PC,TCC1                ;PRINT ERROR AND DROP OFF LINE UNIT
 4749 023440                        ENDIF
      023440                                                          50451$:
 4750                                                           ;REPEAT UNTIL ON LINE OR TIMED OUT.
 4751 023440 004737 017734          JSR PC,NEXTU                ;SET UP FOR NEXT UNIT.
 4752 023444                      ENDDO
      023444 000641                                                   BR      50434$
      023446                                                          50435$:
 4753 023446                    ENDIF
      023446                                                          50433$:
 4754 023446                    IFB PWRFLG EQ #0 THEN
      023446 105737 003547                                            TSTB    PWRFLG
      023452 001026                                                   BNE     50452$
 4755 023454                      MEMORY DATAWT              ;REQUEST MEMORY FROM SUPER FOR RD/WR BUFFERS.
      023454 104431                                                           TRAP    C$MEM
      023456 010037 003414                                                    MOV     R0,DATAWT
 4756 023462                      LET DATARD := DATAWT + #DATCNT          ;SET RD BFR AD
      023462 013737 003414 003416                                     MOV     DATAWT,DATARD
      023470 062737 010000 003416                                     ADD     #DATCNT,DATARD
 4757 023476                      IF @DATAWT LT #DATCNT THEN    ;WHEN NOT ENOUGH FREE MEMO AVAILABLE
      023476 027727 157712 010000                                     CMP     @DATAWT,#DATCNT
      023504 002011                                                   BGE     50453$
 4758 023506                        PRINTF #MEMOM               ;WARN OPERATOR
      023506 012746 023554                                                    MOV     #MEMOM,-(SP)
      023512 012746 000001                                                    MOV     #1,-(SP)
      023516 010600                                                           MOV     SP,R0
      023520 104417                                                           TRAP    C$PNTF
      023522 062706 000004                                                    ADD     #4,SP
 4759 023526                        DOCLN                       ;AND ABORT PASS
      023526 104444                                                           TRAP    C$DCLN
 4760 023530                      ENDIF          ;DIAG MUST BE RE-LOADED IN A CPU WITH LARGER MEMO
      023530                                                          50453$:
 4761 023530                    ENDIF
      023530                                                          50452$:
 4762
 4763 023530                    LET CHGFLG :B= #0              ;CLR CHANGE CMD SEQ TBL FLAG.
      023530 105037 002217                                            CLRB    CHGFLG
 4764 023534                    LET R3 := #ENDFLG
      023534 012703 003546                                            MOV     #ENDFLG,R3
 4765 023540 004737 012110      JSR     PC,CLRERR              ;CLEAR ALL FLAGS.
 4766 023544                    LET PWRFLG :B= #0              ;CLEAR THE POWER FAIL FLAG.
      023544 105037 003547                                            CLRB    PWRFLG
 4767
 4768 023550                    EXIT    INIT
      023550 104432                                                           TRAP    C$EXIT
      023552 000104                                                           .WORD   L10013-.
 4780 023554    045    101    106  MEMOM:  .ASCII  /%AFREE MEMO TOO SMALL FOR RD-WR BFRS%N/
```

```
        023557      122     105     105
        023562      040     115     105
        023565      115     117     040
        023570      124     117     117
        023573      040     123     115
        023576      101     114     114
        023601      040     106     117
        023604      122     040     122
        023607      104     055     127
        023612      122     040     102
        023615      106     122     123
        023620      045     116
4781    023622      045     101     122             .ASCIZ  /%ARE-LOAD IN LARGER MEMO%N/
        023625      105     055     114
        023630      117     101     104
        023633      040     111     116
        023636      040     114     101
        023641      122     107     105
        023644      122     040     115
        023647      105     115     117
        023652      045     116     000
4782
4783                                                .EVEN
4784    023656                                      ENDINIT
        023656                              L10013:
        023656  104411                                                  TRAP    C$INIT
```

```
                                .SBTTL  AUTO DROP SECTION

                                ;++
                                ;SECTION EXECUTED AFTER THE INIT CODE WHEN "ADR" FLAG IS SET BY OPERATOR
                                ;SECTION CHEKS FOR A VALID INTERFACE LOCATION.  DROPS UNIT IF NO RESPONSE
                                ;FROM INTERFACE
                                ;--

                                        BGNAUTO
                                L$AUTO::

                                        JSR PC,FIRSTU                   ;FIND FIRST UNIT
                                        WHILE DEVTBL(R5) NE #END DO     ;
                                                                        50454$:
                                                                        CMP     DEVTBL(R5),#END
                                                                        BEQ     50455$

                                          LET TRAPD4 :B= #0             ;
                                                                        CLRB    TRAPD4
                                          SETVEC #4,#TRAP4,#PRI07       ;SET VECTOR 4
                                                                        MOV     #PRI07,-(SP)
                                                                        MOV     #TRAP4,-(SP)
                                                                        MOV     #4,-(SP)
                                                                        MOV     #3,-(SP)
                                                                        TRAP    C$SVEC
                                                                        ADD     #10,SP

                                          LET R2 := @TSSR(R5)           ;ADDRESS TK25 INTERFACE
                                                                          MOV     @TSSR(R5),R2
                                          CLRVEC #4                     ;CLEAR VECTOR AT 4
                                                                        MOV     #4,R0
                                                                        TRAP    C$CVEC

                                          IFB TRAPD4 NE #0 THEN
                                                                          TSTB    TRAPD4
                                                                        BEQ     50456$

                                            LET FTLCNT(R5) := FTLCNT(R5) + #1
                                                                        INC     FTLCNT(R5)
                                            PRINTF #AUTODM,TSSR(R5)     ;PRINT ERROR
                                                                        MOV     TSSR(R5),-(SP)
                                                                        MOV     #AUTODM,-(SP)
                                                                        MOV     #2,-(SP)
                                                                        MOV     SP,R0
                                                                        TRAP    C$PNTF
                                                                        ADD     #6,SP

                                            LET DROPN := DEVTBL(R5)             ;SAVE # OF UNIT TO BE DROPPED.
                                                                          MOV     DEVTBL(R5),DROPN
                                            LET R0 := R5 SHIFT -1               ;R0=LOGICAL DEVICE NUMBER
                                                                          MOV     R5,R0
                                                                          ASR     R0

                                            DODU R0                     ;DROP THE UNIT: EXEC BGNDU-ENDDU CODE IF IDU = 0
                                                                          TRAP    C$DODU
                                          ELSE
                                                                        BR      50457$
                                                                        50456$:
                                            LET @TSDB(R5) := #GSCPK     ;SEND GET STATUS COMMAND
                                                                          MOV     #GSCPK,@TSDB(R5)
                                            JSR PC,WSSR                 ;WAIT
                                            IF #TS.SSR SETIN @TSSR(R5) THEN
                                                                          BIT     #TS.SSR,@TSSR(R5)
```

```
                     024034  001423
4812                 024036
                     024036  032775  000100  002466          IF #TS.OFL SETIN @TSSR(R5) THEN
                     024044  001416
4813                 024046
                     024046  005265  003320                    LET FTLCNT(R5) := FTLCNT(R5) + #1
4814                 024052
                     024052  016546  002536                    PRINTF #OFLINM,DEVTBL(R5)
                     024056  012746  005223
                     024062  012746  000002
                     024066  010600
                     024070  104417
                     024072  062706  000006
4815                 024076  004737  020120                    JSR PC,DROPUA
4816                 024102                                  ENDIF
                     024102
4817                 024102                                ELSE
                     024102  000416
                     024104
4818                 024104                                  LET FTLCNT(R5) := FTLCNT(R5) + #1
                     024104  005265  003320
4819                 024110                                  PRINTF #NRDYM,DEVTBL(R5)
                     024110  016546  002536
                     024114  012746  024244
                     024120  012746  000002
                     024124  010600
                     024126  104417
                     024130  062706  000006
4820                 024134  004737  020120                    JSR PC,DROPUA
4821                 024140                                  ENDIF
                     024140
4822                 024140                              ENDIF
                     024140
4823                 024140  004737  017734                  JSR PC,NEXTU
4824                 024144                              ENDDO
                     024144  000647
                     024146
4825
4826                 024146                              ENDAUTO
                     024146                        L10014:
                     024146  104461
4827
4828                 024150     045     101     102  AUTODM: .ASCII  /%ABUS TRAP AT %O6%N/
                     024153     125     123     040
                     024156     124     122     101
                     024161     120     040     101
                     024164     124     040     045
                     024167     117     066     045
                     024172     116
4829                 024173     045     101     111          .ASCIZ  /%AINTERFACE BAD OR NOT SET TO ABOVE AD%N/
                     024176     116     124     105
                     024201     122     106     101
                     024204     103     105     040
                     024207     102     101     104
                     024212     040     117     122
                     024215     040     116     117
                     024220     124     040     123
```

```
BEQ     50460$

BIT     #TS.OFL,@TSSR(R5)
BEQ     50461$

INC     FTLCNT(R5)

        MOV     DEVTBL(R5),-(SP)
        MOV     #OFLINM,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTF
        ADD     #6,SP

50461$:

BR      50462$
50460$:

INC     FTLCNT(R5)

        MOV     DEVTBL(R5),-(SP)
        MOV     #NRDYM,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTF
        ADD     #6,SP

50462$:

50457$:

BR      50454$
50455$:

        TRAP    C$AUTO
```

```
        024223      105     124     040
        024226      124     117     040
        024231      101     102     117
        024234      126     105     040
        024237      101     104     045
        024242      116     000
 4830   024244      045     101     125  NRDYM:  .ASCIZ  /%AUNIT %D1%A NOT RDY%N/
        024247      116     111     124
        024252      040     045     104
        024255      061     045     101
        024260      040     116     117
        024263      124     040     122
        024266      104     131     045
        024271      116     000
 4831                                         .EVEN
 4832
 4833                                 ;       DEVICE BUS TRAP HANDLER
 4834                                 ;       OUTPUT: TRAPD4 BYTE  1: TRAPED AT 4
 4835                                 ;                            0: NO TRAP
 4836
 4837   024274                        TRAP4:: LET TRAPD4 :B= TRAPD4 + #1
        024274  105237  003550                                                  INCB    TRAPD4
 4838   024300  000002                        RTI
 4839
 4840
 4841
```

```
                                .SBTTL  CLEANUP CODING SECTION

                                ;**
                                ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
                                ; AT THE END OF EACH PASS.
                                ;--

                                        BGNCLN
                                L$CLEAN::
                                        IF LCSR NE #0 THEN
024302  005737  003464                                                    TST     LCSR
024306  001403                                                            BEQ     50463$
                                        LET @LCSR := @LCSR CLR.BY #100  ;SHUT OFF THE INTERRUPTS
024310  042777  000100  157146                                           BIC     #100,@LCSR
                                        ENDIF
024316                                                                  50463$:

                                        JSR     PC,FIRSTU               ;FIND FIRST UNIT.
                                        WHILE DEVTBL(R5) NE #END DO
                                                                        50464$:
024322  026527  002536  177777                                          CMP     DEVTBL(R5),#END
024330  001410                                                            BEQ     50465$
                                          JSR PC,WSSR                   ;WAIT FOR UNIT READY OR TIMEOUT.
                                          CLRVEC        TSVCT(R5)       ;RELEASE INTERRUPT VECTORS FOR ALL DEV.
024336  016500  002476                                                    MOV     TSVCT(R5),R0
024342  104436                                                            TRAP    C$CVEC
                                          JSR   PC,NEXTU                ;FIND NEXT UNIT.
                                        ENDDO
024350  000764                                                            BR      50464$
                                                                        50465$:

                                        EXIT    CLN
024352  104432                                                          TRAP    C$EXIT
024354  000002                                                          .WORD   L10015-.
                                        .EVEN
                                        ENDCLN
                                L10015:
024356  104412                                                          TRAP    C$CLEAN
```

```
                                .SBTTL  DROP UNIT SECTION

                                ;++
                                ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
                                ; TO NO LONGER BE TESTED.  THAT CODE SHALL BE EXECUTED WHEN DODU
                                ;MACRO IS CALLED WHILE IDU FLAG IS NOT SET BY OPERATOR
                                ;--

024360                                  BGNDU
024360                          L$DU::

                                        LET R5 := R0 SHIFT 1            ;R5 = LOGICAL DEVICE NUMBER X 2.
024360  010005                                                                 MOV     R0,R5
024362  006305                                                                 ASL     R5
                                        LET DEVTBL(R5) := #NINUSE       ;SET NOT IN USE FLAG FOR THE DEVICE.
024364  012765  177774  002536                                                 MOV     #NINUSE,DEVTBL(R5)
                                        PRINTF  #DROPDM,DROPN           ;PRINT DROP DEVICE MESSAGE
024372  013746  020210                                                         MOV     DROPN,-(SP)
024376  012746  004712                                                         MOV     #DROPDM,-(SP)
024402  012746  000002                                                         MOV     #2,-(SP)
024406  010600                                                                 MOV     SP,R0
024410  104417                                                                 TRAP    C$PNTF
024412  062706  000006                                                         ADD     #6,SP
                                        EXIT    DU
024416  000167                                                                 .WORD   J$JMP
024420  000000                                                                 .WORD   L10016-2-.
                                        .EVEN

                                        ENDDU
024422                          L10016:
024422  104453                                                                 TRAP    C$DU
```

SEQ 0145

```
                                .SBTTL  ADD UNIT SECTION

                                ;+*
                                ; THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
                                ;  TO BE (A) TESTED FOR THE FIRST TIME, OR (B) RESUMED IN TESTING.  IF
                                ;  "EF.AUNIT" IS SET, THE UNIT WILL BE TESTED AS A NEW UNIT.
                                ;--

                                        BGNAU
                                L$AU::

                                        LET R5 := R0 SHIFT 1            ;R5 = LOGICAL DEVICE NUMBER X 2.
                                                                                MOV     R0,R5
                                                                                ASL     R5
                                        LET DEVTBL(R5) := R0    ;STORE UNIT # IN DEVICE TABLE.
                                                                                MOV     R0,DEVTBL(R5)
                                        GPHARD  R0,R0                   ;GET HARDWARE P TABLE FROM SUPER.
                                                                                TRAP    C$GPHRD
                                        LET TSSR(R5) := (R0)            ;SAVE TSSR ADDRESS.
                                                                                MOV     (R0),TSSR(R5)
                                        LET TSDB(R5) := (R0)+ - #2      ;SAVE TSDB ADDRESS.
                                                                                MOV     (R0)+,TSDB(R5)
                                                                                SUB     #2,TSDB(R5)
                                        LET TSVCT(R5) := (R0)           ;SAVE INTERRUPT VECTOR ADDRESS.
                                                                                MOV     (R0),TSVCT(R5)
                                        SETVEC TSVCT(R5),TS4INT(R5),#INTPRI     ;SET UP INTERUPT PROCESSIN  CONDITIONS.
                                                                                MOV     #IN       (SP)
                                                                                MOV     TS4I      .-(SP)
                                                                                MOV     TSVCT(    SP)
                                                                                MOV     #3,-(SP.
                                                                                TRAP    C$SVEC
                                                                                ADD     #10,SP
                                        LET INTFLG(R5) := #0            ;CLEAR INTERRUPT FLAGS.
                                                                                CLR     INTFLG(R5)

                                        EXIT    AU

                                                                                .WORD   J$JMP
                                                                                .WORD   L10017-2-.

                                        .EVEN

                                        ENDAU
                                L10017:
                                                                                TRAP    C$AU

                                        ENDMOD
```

```
 4966
 4977                                   .TITLE HARDWARE TESTS
 4978
 4979                                   .SBTTL  TEST 1:  BASIC FUNCTIONS.
 4980
 4981                                   ;++
 4982                                   ; TEST TO EXECUTE ALL TK25 FUNCTIONS.
 4983                                   ;--
 4984
 4985 024520                                       BGNMOD
 4986
 4987 024520                                       BGNTST
      024520                            T1::
 4988
 4989 024520                                       LET RANDOM :B= #0                ;CLR THE RANDOM OPERATIONS FLAG.
      024520  105037  003533                                                                CLRB    RANDOM
 4990 024524                                       LET EXPBOT :B= #0                ;CLR EXPECT BOT FLAG.
      024524  105037  003532                                                                CLRB    EXPBOT
 4991
 4992 024530                                       BGNSUB                           ;SUBTEST 1 - SET CHAR, DRIVE INIT, GET STATUS.
      024530                            T1.1:
      024530  104402                                                                       TRAP    C$BSUB
 4993
 4994 024532                                       LET R2 := #BFSEQ0                ;ADR OF CMD SEQ.
      024532  012702  025356                                                                MOV     #BFSEQ0,R2
 4995 024536  004737  025332                       JSR     PC,BFSEQ                 ;SET UP CMD SEQ.
 4996 024542  004737  007036                       JSR     PC,EXALL                 ;EXECUTE CMD SEQ ON ALL DEVICES.
 4997 024546  004737  017666                       JSR     PC,FIRSTU                ;FIND THE FIRST UNIT.
 4998 024552                                       WHILE DEVTBL(R5) NE #END DO      ;WHILE THERE ARE MORE DEVICES:
      024552                                                                           50466$:
      024552  026527  002536  177777                                                    CMP     DEVTBL(R5),#END
      024560  001434                                                                    BEQ     50467$
 4999 024562                                         LET R2 := MSGPKA(R5)           ;GET MSG PACKET ADR,
      024562  016502  002506                                                           MOV     MSGPKA(R5),R2
 5000 024566                                         LET R2 := R2 + #12             ;GET XSTAT2 ADR,
      024566  062702  000012                                                           ADD     #12,R2
 5001 024572                                         LET TS4CL(R5) := (R2) CLR.BY #177400  ;STORE CODE LEVEL FROM DTR BYTE,
      024572  011265  002526                                                           MOV     (R2),TS4CL(R5)
      024576  042765  177400  002526                                                   BIC     #177400,TS4CL(R5)
 5002 024604                                         IF PASCNT(R5) EQ #1 THEN       ;IF THIS IS PASS 1 THEN:
      024604  026527  003260  000001                                                   CMP     PASCNT(R5),#1
      024612  001014                                                                  BNE     50470$
 5003 024614                                           PRINTF #CODELM,DEVTBL(R5),TS4CL(R5) ;PRINT THE TK25 MICROCODE LEVEL.
      024614  016546  002526                                                                  MOV     TS4CL(R5),-(SP)
      024620  016546  002536                                                                  MOV     DEVTBL(R5),-(SP)
      024624  012746  004066                                                                  MOV     #CODELM,-(SP)
      024630  012746  000003                                                                  MOV     #3,-(SP)
      024634  010600                                                                         MOV     SP,R0
      024636  104417                                                                         TRAP    C$PNTF
      024640  062706  000010                                                                 ADD     #10,SP
 5004 024644                                         ENDIF
      024644                                                                           50470$:
 5005 024644  004737  017734                         JSR   PC,NEXTU                 ;FIND NEXT UNIT.
 5006 024650                                       ENDDO
      024650  000740                                                                  BR      50466$
      024652                                                                           50467$:
 5007 024652                                       ENDSUB
```

```
 024652                   L10021:
 024652  104403                                                                   TRAP    C$ESUB

 024654                            BGNSUB                         ;SUBTEST 2 - REWIND.
 024654                   T1.2:
 024654  104402                                                                   TRAP    C$BSUB

 024656                            LET R2 := #BFSEQ1              ;ADR OF CMD SEQ.
 024656  012702  025430                                                    MOV     #BFSEQ1,R2
 024662  004737  025332            JSR     PC,BFSEQ               ;SET UP CMD SEQ.
 024666  004737  007036            JSR     PC,EXALL               ;EXECUTE CMD SEQ ON ALL DEVICES.
 024672                            LET STAFLG :B= #0              ;CLEAR START FLAG
 024672  105037  003546                                                    CLRB    STAFLG
 024676                            ENDSUB
 024676                   L10022:
 024676  104403                                                                   TRAP    C$ESUB

 024700                            BGNSUB                         ;SUBTEST 3 - WRITE/VERIFY.
 024700                   T1.3:
 024700  104402                                                                   TRAP    C$BSUB

 024702                            LET R2 := #BFSEQ2              ;ADR OF CMD SEQ.
 024702  012702  025442                                                    MOV     #BFSEQ2,R2
 024706  004737  025332            JSR     PC,BFSEQ               ;SET UP CMD SEQ.
 024712  004737  007036            JSR     PC,EXALL               ;EXECUTE CMD SEQ ON ALL DEVICES.
 024716                            ENDSUB
 024716                   L10023:
 024716  104403                                                                   TRAP    C$ESUB

 024720                            BGNSUB                         ;SUBTEST 4 - WRITE TAPE MARK, ERASE.
 024720                   T1.4:
 024720  104402                                                                   TRAP    C$BSUB

 024722                            LET R2 := #BFSEQ3              ;ADR OF CMD SEQ.
 024722  012702  025534                                                    MOV     #BFSEQ3,R2
 024726  004737  025332            JSR     PC,BFSEQ               ;SET UP CMD SEQ.
 024732  004737  007036            JSR     PC,EXALL               ;EXECUTE CMD SEQ ON ALL DEVICES.
 024736                            ENDSUB
 024736                   L10024:
 024736  104403                                                                   TRAP    C$ESUB

 024740                            BGNSUB                         ;SUBTEST 5 - SPACE FILES.
 024740                   T1.5:
 024740  104402                                                                   TRAP    C$BSUB

 024742                            LET R2 := #BFSEQ4              ;ADR OF CMD SEQ.
 024742  012702  025606                                                    MOV     #BFSEQ4,R2
 024746  004737  025332            JSR     PC,BFSEQ               ;SET UP CMD SEQ.
 024752  004737  007036            JSR     PC,EXALL               ;EXECUTE CMD SEQ ON ALL DEVICES.
 024756                            ENDSUB
 024756                   L10025:
 024756  104403                                                                   TRAP    C$ESUB

 024760                            BGNSUB                         ;SUBTEST 6 - SPACE RECORDS.
 024760                   T1.6:
 024760  104402                                                                   TRAP    C$BSUB
```

```
5040 024762                            LET R2 := #BFSEQ5          ;ADR OF CMD SEQ.
     024762  012702  025650                                              MOV     #BFSEQ5,R2
5041 024766  004737  025332            JSR     PC,BFSEQ           ;SET UP CMD SEQ.
5042 024772  004737  007036            JSR     PC,EXALL           ;EXECUTE CMD SEQ ON ALL DEVICES.
5043 024776                            ENDSUB
     024776                        L10026:
     024776  104403                                                TRAP    C$ESUB
5044
5045 025000                            BGNSUB                     ;SUBTEST 7 - WRITE RETRY.
     025000                        T1.7:
     025000  104402                                                TRAP    C$BSUB
5046
5047 025002                            LET R2 := #BFSEQ6          ;ADR OF CMD SEQ.
     025002  012702  025722                                              MOV     #BFSEQ6,R2
5048 025006  004737  025332            JSR     PC,BFSEQ           ;SET UP CMD SEQ.
5049 025012  004737  007036            JSR     PC,EXALL           ;EXECUTE CMD SEQ ON ALL DEVICES.
5050 025016                            ENDSUB
     025016                        L10027:
     025016  104403                                                TRAP    C$ESUB
5051
5052 025020                            BGNSUB                     ;SUBTEST 8 - READ REV RETRY.
     025020                        T1.8:
     025020  104402                                                TRAP    C$BSUB
5053
5054 025022                            LET R2 := #BFSEQ7          ;ADR OF CMD SEQ.
     025022  012702  025754                                              MOV     #BFSEQ7,R2
5055 025026  004737  025332            JSR     PC,BFSEQ           ;SET UP CMD SEQ.
5056 025032  004737  007036            JSR     PC,EXALL           ;EXECUTE CMD SEQ ON ALL DEVICES.
5057 025036                            ENDSUB
     025036                        L10030:
     025036  104403                                                TRAP    C$ESUB
5058
5059 025040                            BGNSUB                     ;SUBTEST 9 - READ FWD RETRY.
     025040                        T1.9:
     025040  104402                                                TRAP    C$BSUB
5060
5061 025042                            LET R2 := #BFSEQ8          ;ADR OF CMD SEQ.
     025042  012702  026006                                              MOV     #BFSEQ8,R2
5062 025046  004737  025332            JSR     PC,BFSEQ           ;SET UP CMD SEQ.
5063 025052  004737  007036            JSR     PC,EXALL           ;EXECUTE CMD SEQ ON ALL DEVICES.
5064 025056                            ENDSUB
     025056                        L10031:
     025056  104403                                                TRAP    C$ESUB
5065
5066 025060                            BGNSUB                     ;SUBTEST 10- CLEAN.
     025060                        T1.10:
     025060  104402                                                TRAP    C$BSUB
5067
5068 025062                            LET R2 := #BFSEQ9          ;ADR OF CMD SEQ.
     025062  012702  026040                                              MOV     #BFSEQ9,R2
5069 025066  004737  025332            JSR     PC,BFSEQ           ;SET UP CMD SEQ.
5070 025072  004737  007036            JSR     PC,EXALL           ;EXECUTE CMD SEQ ON ALL DEVICES.
5071 025076                            ENDSUB
     025076                        L10032:
     025076  104403                                                TRAP    C$ESUB
5072
5073 025100                            BGNSUB                     ;SUBTEST 11 - WTV SWAPPED DATA BYTES.
```

```
                025100                          T1.11:
                025100  104402                                                  TRAP    C$BSUB
5074
5075 025102                             LET R2 := #BFSE10               ;ADR OF CMD SEQ.
                025102  012702  026062                                          MOV     #BFSE10,R2
5076 025106     004737  025332          JSR     PC,BFSEQ                ;SET UP CMD SEQ.
5077 025112     004737  007036          JSR     PC,EXALL                ;WRITE/VERIFY RECORDS 1 AND 2.
5078 025116                             LET SWBFLG :B= #1               ;ENABLE BYTE SWAPPING.
                025116  112737  000001  003536                                  MOVB    #1,SWBFLG
5079 025124     004737  007036          JSR     PC,EXALL                ;WRITE/VERIFY RECORDS 3 AND 4.
5080 025130                             LET SWBFLG :B= #0               ;DISABLE BYTE SWAPPING.
                025130  105037  003536                                          CLRB    SWBFLG
5081 025134                             ENDSUB
                025134                  L10033:
                025134  104403                                                  TRAP    C$ESUB
5082
5083 025136                             LET R2 := DATAWT + #10.         ;INIT WRITE BUFFER POINTER.
                025136  013702  003414                                          MOV     DATAWT,R2
                025142  062702  000012                                          ADD     #10.,R2
5084 025146                             WHILE R2 NE DATAWT DO   ;UNTIL 10 BYTES HAVE BEEN SWAPPED.
                025146                                                  50471$:
                025146  020237  003414                                          CMP     R2,DATAWT
                025152  001402                                                  BEQ     50472$
5085 025154     000342                    SWAB -(R2)                    ;SWAP DATA BYTES IN WRITE BUFFER.
5086 025156                             ENDDO
                025156  000773                                                  BR      50471$
                025160                                                  50472$:
5087 025160                             LET T1SWB :B= T1SWB + #1        ;SET T1 SWAP BYTES FLAG FOR "CKDATA" SUBR
                025160  105237  003542                                          INCB    T1SWB
5088
5089 025164                             BGNSUB                          ;SUBTEST  2 - READ SWAPPED DATA BYTES.
                025164                  T1.12:
                025164  104402                                                  TRAP    C$BSUB
5090
5091 025166                             LET CMDWRD := #RDR              ;CMD IS READ REV.
                025166  012737  104401  003426                                  MOV     #RDR,CMDWRD
5092 025174     004737  016642          JSR     PC,VFEXC                ;VERIFY ODD LENGTH SWAP (RECORD 4).
5093 025200                             LET CMDPKT+CP.CNT := #12        ;CHANGE BYTE COUNT TO 10.
                025200  012737  000012  002322                                  MOV     #12,CMDPKT+CP.CNT
5094 025206     004737  016642          JSR     PC,VFEXC                ;VERIFY EVEN LENGTH SWAP (RECORD 3).
5095 025212                             LET SWBFLG :B= #1               ;ENABLE BYTE SWAPPING.
                025212  112737  000001  003536                                  MOVB    #1,SWBFLG
5096 025220                             LET CMDPKT+CP.CNT := #11        ;CHANGE BYTE COUNT TO 9.
                025220  012737  000011  002322                                  MOV     #11,CMDPKT+CP.CNT
5097 025226     004737  016642          JSR     PC,VFEXC                ;VERIFY ODD LENGTH SWAP (RECORD 2).
5098 025232                             LET CMDPKT+CP.CNT := #12        ;CHANGE BYTE COUNT TO 10.
                025232  012737  000012  002322                                  MOV     #12,CMDPKT+CP.CNT
5099 025240     004737  016642          JSR     PC,VFEXC                ;VERIFY EVEN LENGTH SWAP (RECORD 1).
5100 025244                             LET CMDWRD := #RDF              ;CMD IS READ FWD.
                025244  012737  104001  003426                                  MOV     #RDF,CMDWRD
5101 025252     004737  016642          JSR     PC,VFEXC                ;VERIFY EVEN LENGTH SWAP (RECORD 1).
5102 025256                             LET CMDPKT+CP.CNT := #11        ;CHANGE BYTE COUNT TO 9.
                025256  012737  000011  002322                                  MOV     #11,CMDPKT+CP.CNT
5103 025264     004737  016642          JSR     PC,VFEXC                ;VERIFY ODD LENGTH SWAP (RECORD 2).
5104 025270                             LET SWBFLG :B= #0               ;DISABLE BYTE SWAPPING.
                025270  105037  003536                                          CLRB    SWBFLG
5105 025274                             LET CMDPKT+CP.CNT := #12        ;CHANGE BYTE COUNT TO 10.
```

```
     025274  012737  000012  002322                                               MOV     #12,CMDPKT+CP.CNT
5106 025302  004737  016642                  JSR     PC,VFEXC                ;VERIFY EVEN LENGTH SWAP (RECORD 3).
5107 025306                                  LET CMDPKT+CP.CNT := #11        ;CHANGE BYTE COUNT TO 9.
     025306  012737  000011  002322                                               MOV     #11,CMDPKT+CP.CNT
5108 025314  004737  016642                  JSR     PC,VFEXC                ;VERIFY ODD LENGTH SWAP (RECORD 4).
5109
5110 025320                                  ENDSUB
     025320                         L10034:
     025320  104403                                                                       TRAP    C$ESUB
5111
5112 025322                                  LET T1SWB :B= #0                ;CLEAR T1 SWAP BYTES FLAG
     025322  105037  003542                                                       CLRB    T1SWB
5113
5114
5115 025326                                  EXIT    TST
     025326  104432                                                                       TRAP    C$EXIT
     025330  000554                                                                       .WORD   L10020-.
```

```
                                ;       SUBROUTINE TO MOVE A COMMAND SEQUENCE TO THE SEQUENCE TABLE.
                                ;       INPUTS:         R2 = FWA OF COMMAND SEQUENCE.
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

025332                          BFSEQ:  LET R1 := #CMDSEQ               ;INIT SEQ TABLE ADDRESS.
025332  012701  003554                                                          MOV     #CMDSEQ,R1
                                        WHILE (R2) NE #END DO           ;WHILE THERE ARE MORE COMMANDS:
025336                                                                          50473$:
025336  021227  177777                                                          CMP     (R2),#END
025342  001402                                                                  BEQ     50474$
                                          LET (R1)+ := (R2)+            ;MOVE COMMANDS TO SEQ TABLE.
025344  012221                                                                  MOV     (R2)+,(R1)+
                                        ENDDO
025346  000773                                                                  BR      50473$
025350                                                                          50474$:
                                        LET (R1) := #END                ;STORE END OF SEQUENCE CODE.
025350  012711  177777                                                          MOV     #END,(R1)
025354  000207                        RTS     PC                      ;RETURN.


                                ;       BASIC FUNCTION COMMAND SEQUENCE

025356  140004                  BFSEQ0: .WORD   SCH                     ;SET CHAR. 200.         (1)
025360  000200                                  200
025362  000001                                  1
025364  000000                                  0
025366  100013                                  DRI                     ;DRIVE INIT.            (2)
025370  000001                                  1
025372  000001                                  1
025374  000000                                  0
025376  140004                                  SCH                     ;SET CHAR. 20           (3)
025400  000020                                  20
025402  000001                                  1
025404  000000                                  0
025406  100017                                  GES                     ;GET STATUS.            (4)
025410  000001                                  1
025412  000001                                  1
025414  000000                                  0
025416  140004                                  SCH                     ;SET CHAR. 40.          (5)
025420  000040                                  40
025422  000001                                  1
025424  000000                                  0
025426  177777                          .WORD   END

025430  102010                  BFSEQ1:         RWD                     ;REWIND TWICE.          (6)
025432  000001                                  1
025434  000002                                  2
025436  000000                                  0
025440  177777                          .WORD   END

025442  104105                  BFSEQ2:         WTV                     ;WRITE/VERIFY PAT 1.    (7)
025444  010000                                  DATCNT
025446  000001                                  1
025450  000001                                  1
```

```
5166 025452  104105                        WTV             ;WTV PAT 2.          (8)
5167 025454  010000                        DATCNT
5168 025456  000001                        1
5169 025460  000002                        2
5170 025462  104105                        WTV             ;WTV PAT 3.          (9)
5171 025464  010000                        DATCNT
5172 025466  000001                        1
5173 025470  000003                        3
5174 025472  104105                        WTV             ;WTV PAT 4.          (10)
5175 025474  010000                        DATCNT
5176 025476  000001                        1
5177 025500  000004                        4
5178 025502  104105                        WTV             ;WTV PAT 5.          (11)
5179 025504  010000                        DATCNT
5180 025506  000001                        1
5181 025510  000005                        5
5182 025512  104105                        WTV             ;WTV PAT 6.          (12)
5183 025514  010000                        DATCNT
5184 025516  000001                        1
5185 025520  000006                        6
5186 025522  104105                        WTV             ;WTV PAT 0.          (13)
5187 025524  010000                        DATCNT
5188 025526  000001                        1
5189 025530  000000                        0
5190 025532  177777                .WORD   END
5191
5192 025534  100011           BFSEQ3:      WTM             ;WRITE TAPE MARK.    (14)
5193 025536  000001                        1
5194 025540  000001                        1
5195 025542  000000                        0
5196 025544  104005                        WRT             ;WRITE 10 RECORDS.   (15)
5197 025546  010000                        DATCNT
5198 025550  000010                        10
5199 025552  000001                        1
5200 025554  100411                        ERS             ;ERASE 10 TIMES.     (16)
5201 025556  000001                        1
5202 025560  000010                        10
5203 025562  000000                        0
5204 025564  100011                        WTM             ;WRITE TAPE MARK.    (17)
5205 025566  000001                        1
5206 025570  000001                        1
5207 025572  000000                        0
5208 025574  101011                        WTR             ;WTM RETRY           (18)
5209 025576  000001                        1
5210 025600  000001                        1
5211 025602  000000                        0
5212 025604  177777                .WORD   END
5213
5214 025606  105410           BFSEQ4:      SFR             ;SPACE 2 FILES REV.  (19)
5215 025610  000002                        2
5216 025612  000001                        1
5217 025614  000000                        0
5218 025616  105010                        SFF             ;SPACE 2 FILES FWD.  (20)
5219 025620  000002                        2
5220 025622  000001                        1
5221 025624  000000                        0
5222 025626  105410                        SFR             ;SPACE 2 FILES REV.  (21)
```

```
5223 025630  000001                              1
5224 025632  000002                              2
5225 025634  000000                              0
5226 025636  105010                              SFF             ;SPACE 2 FILES FWD.     (22)
5227 025640  000001                              1
5228 025642  000002                              2
5229 025644  000000                              0
5230 025646  177777                      .WORD   END
5231
5232 025650  102010              BFSEQ5:         RWD             ;REWIND.                (23)
5233 025652  000001                              1
5234 025654  000001                              1
5235 025656  000000                              0
5236 025660  104010                              SRF             ;SPACE 7 RECORDS FWD.   (24)
5237 025662  000007                              7
5238 025664  000001                              1
5239 025666  000000                              0
5240 025670  104410                              SRR             ;SPACE 7 RECORDS REV.   (25)
5241 025672  000007                              7
5242 025674  000001                              1
5243 025676  000000                              0
5244 025700  104010                              SRF             ;SPACE 7 RECORDS FWD.   (26)
5245 025702  000001                              1
5246 025704  000007                              7
5247 025706  000000                              0
5248 025710  104410                              SRR             ;SPACE 7 RECORDS REV.   (27)
5249 025712  000001                              1
5250 025714  000007                              7
5251 025716  000000                              0
5252 025720  177777                      .WORD   END
5253
5254 025722  102010              BFSEQ6:         RWD             ;REWIND.                (28)
5255 025724  000001                              1
5256 025726  000001                              1
5257 025730  000000                              0
5258 025732  104005                              WRT             ;WRITE.                 (29)
5259 025734  010000                              DATCNT
5260 025736  000001                              1
5261 025740  000001                              1
5262 025742  105005                              WRR             ;WRITE RETRY.           (30)
5263 025744  010000                              DATCNT
5264 025746  000001                              1
5265 025750  000001                              1
5266 025752  177777                      .WORD   END
5267
5268 025754  104401              BFSEQ7:         RDR             ;READ REV.              (31)
5269 025756  010000                              DATCNT
5270 025760  000001                              1
5271 025762  000001                              1
5272 025764  105401                              RNR             ;READ NEXT REV.         (32)
5273 025766  010000                              DATCNT
5274 025770  000001                              1
5275 025772  000001                              1
5276 025774  125401                              RNF             ;READ NEXT FWD.         (33)
5277 025776  010000                              DATCNT
5278 026000  000001                              1
5279 026002  000001                              1
```

```
5280 026004  177777                    .WORD   END
5281
5282 026006  104001          BFSEQ8:           RDF                 ;READ FWD.          (34)
5283 026010  010000                            DATCNT
5284 026012  000001                            1
5285 026014  000001                            1
5286 026016  105001                            RPF                 ;READ PREVIOUS FWD. (35)
5287 026020  010000                            DATCNT
5288 026022  000001                            1
5289 026024  000001                            1
5290 026026  125001                            RPR                 ;READ PREVIOUS REV. (36)
5291 026030  010000                            DATCNT
5292 026032  000001                            1
5293 026034  000001                            1
5294 026036  177777                    .WORD   END
5295
5296 026040  101012          BFSEQ9:   .WORD   CLN                 ;CLEAN.             (37)
5297 026042  000001                            1
5298 026044  000001                            1
5299 026046  000000                            0
5300 026050  102010                            RWD                 ;REWIND             (38)
5301 026052  000001                            1
5302 026054  000001                            1
5303 026056  000000                            0
5304 026060  177777                    .WORD   END                 ;END OF SEQUENCE.
5305
5306 026062  104105          BFSE10:           WTV                 ;WRITE/VERIFY EVEN LENGTH.  (39)
5307 026064  000012                            12
5308 026066  000001                            1
5309 026070  000000                            0
5310 026072  104105                            WTV                 ;WRITE/VERIFY ODD LENGTH.   (40)
5311 026074  000011                            11
5312 026076  000001                            1
5313 026100  000000                            0
5314 026102  177777                    .WORD   END
5315                                   .EVEN
5316
5317 026104                            ENDTST
     026104                  L10020:
     026104  104401                                TRAP    C$ETST
```

```
5319                                    .SBTTL  TEST 2: DATA RELIABILITY.
5320
5321                                    ;++
5322                                    ; TEST TO CHECK THE DATA RELIABILITY OF THE TK25.
5323                                    ;--
5324 026106                                     BGNTST
     026106                             T2::
5325
5326 026106                                     LET RANDOM :B= #1               ;SET THE RANDOM OPERATIONS FLAG.
     026106  112737  000001  003533                                                     MOVB    #1,RANDOM
5327 026114                                     LET EXPBOT :B= #0               ;CLEAR EXPECT BOT FLAG.
     026114  105037  003532                                                     CLRB    EXPBOT
5328 026120                                     LET R2 := #DATCNT - #1          ;SET UP THE RECORD LENGTH MASK,
     026120  012702  010000                                                     MOV     #DATCNT,R2
     026124  005302                                                             DEC     R2
5329 026126                                     LET LENMSK := COMP R2           ;ALLOW MAXIMUM BUFFER.
     026126  010237  003436                                                     MOV     R2,LENMSK
     026132  005137  003436                                                     COM     LENMSK
5330 026136  004737  006772                     JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
5331 026142                                     IFB STAFLG NE #0 THEN           ;IF STARTING THEN:
     026142  105737  003546                                                     TSTB    STAFLG
     026146  001404                                                             BEQ     50475$
5332 026150  004737  007016                       JSR   PC,SETRW                ;CMD2=REWIND
5333 026154                                       LET STAFLG :B= #0             ;CLR START FLAG.
     026154  105037  003546                                                     CLRB    STAFLG
5334 026160                                     ENDIF
     026160                                                                     50475$:
5335 026160                                     LET (R1)+ := #WRT               ;CMD3 = WRITE.
     026160  012721  104005                                                     MOV     #WRT,(R1)+
5336 026164                                     LET (R1)+ := #DATCNT            ;SET BRF TO MAX FOR PATTERN GENERATION.
     026164  012721  010000                                                     MOV     #DATCNT,(R1)+
5337 026170                                     LET R2 := COMP #RNOPSC
     026170  012702  177740                                                     MOV     #RNOPSC,R2
     026174  005102                                                             COM     R2
5338 026176                                     LET (R1)+ := R2                 ;31 OPERATIONS.
     026176  010221                                                             MOV     R2,(R1)+
5339 026200                                     LET (R1)+ := #RANP              ;RANDOM PATTERN.
     026200  012721  000007                                                     MOV     #RANP,(R1)+
5340 026204                                     REPEAT                          ;REPEAT TO EOT:
     026204                                                                     50476$:
5341 026204                                       WHILE R1 LT #SEQEND DO        ;FILL SEQ TBL WITH RANDOM CMDS.
     026204                                                                     50477$:
     026204  020127  003644                                                     CMP     R1,#SEQEND
     026210  002012                                                             BGE     50500$
5342 026212
     026212  063737  003440  003442                 LET RANS := RANS + RANB                 ADD     RANB,RANS
5343 026220                                       LET R2 := RANS CLR.BY #177741  ;R2 = RANDOM # (0 - 36).
     026220  013702  003442                                                     MOV     RANS,R2
     026224  042702  177741                                                     BIC     #177741,R2
5344 026230  004772  026366                       JSR PC,@RANCMD(R2)            ;SET UP A RANDOM CMD + BRF.
5345 026234                                     ENDDO
     026234  000763                                                             BR      50477$
     026236                                                                     50500$:
5346 026236                                     LET (R1) := #END                ;STORE END OF SEQUENCE CODE IN TABLE.
     026236  012711  177777                                                     MOV     #END,(R1)
5347 026242  004737  007036                     JSR   PC,EXALL                  ;GO EXECUTE ALL CMDS IN SEQUENCE TABLE.
5348 026246                                     LET R1 := #CMDSEQ               ;INIT CMD SEQ TBL POINTER.
```

```
                026246  012701  003554                                                  MOV     #CMDSEQ,R1
5349            026252                          UNTIL R2 NE #0                          ;REPEAT UNTIL EOT IS REACHED
                026252  005702                                                          TST     R2
                026254  001753                                                          BEQ     50476$
5350            026256                          LET ALLEOT :B= ALLEOT + #1              ;FLAG ALL UNITS @ EOT
                026256  105237  003543                                                  INCB    ALLEOT
5351            026262  000240                  NOP
5352            026264  000240                  NOP
5353            026266  000240                  NOP
5354            026270  004737  030364          JSR     PC,T5WEOT                       ;WRITE ONE RECORD BEYOND EOT ON ALL UNITS
5355                                                                                    ;SO THAT SHORTER READ STOP DISTANCE
5356                                                                                    ;SHALL POSITION HEAD IN CLEAN IRG GAP
5357                                                                                    ;READ REV THAT EXTRA REC TO RE-POSITION THE TAPE
5358            026274  004737  026426          JSR     PC,RANRD                        ;SET UP READ REV/FWD CMDS.
5359            026300                          LET CMDSEQ+4 := COMP #RNOPSC            ;# OF RECORDS FOR READ REV.
                026300  012737  177740  003560                                          MOV     #RNOPSC,CMDSEQ+4
                026306  005137  003560                                                  COM     CMDSEQ+4
5360            026312                          LET CMDSEQ+14 := CMDSEQ+4               ;# OF RECORDS FOR READ FORWARD.
                026312  013737  003560  003570                                          MOV     CMDSEQ+4,CMDSEQ+14
5361            026320                          LET (R1) := #END                        ;STORE END OF SEQUENSE CODE IN SEQ TABLE.
                026320  012711  177777                                                  MOV     #END,(R1)
5362            026324  004737  007036          JSR     PC,EXALL                        ;GO EXECUTE READ REV/FWD OF LAST N RECORDS.
5363            026330                          LET ALLEOT :B= #0                       ;CLEAR ALL UNITS @ EOT FLAG
                026330  105037  003543                                                  CLRB    ALLEOT
5364            026334                          LET RPTFLG :B= #1                       ;REQUEST PERFORMANCE REPORT DURING REWIND.
                026334  112737  000001  003535                                          MOVB    #1,RPTFLG
5365            026342                          LET R1 := #CMDSEQ                       ;INIT SEQ TBL POINTER,
                026342  012701  003554                                                  MOV     #CMDSEQ,R1
5366            026346  004737  007016          JSR PC,SETRW                            ;STORE REWIND IN SEQ TBL.
5367            026352                          LET (R1) := #END                        ;STORE END IN SEQ TBL,
                026352  012711  177777                                                  MOV     #END,(R1)
5368            026356  004737  007036          JSR PC,EXALL                            ;EXECUTE REWIND CMD ON ALL UNITS
5369
5370            026362                          EXIT    TST
                026362  104432                                                          TRAP    C$EXIT
                026364  000174                                                          .WORD   L10035-.
5371
```

```
5373                                    ;       ADDRESSES OF SUBROUTINES USED TO SET UP RANDOM OPERATIONS IN
5374                                    ;       THE DATA RELIABILITY TEST.
5375
5376 026366  026502                     RANCMD: RANWR                   ;WRITE.
5377 026370  026502                             RANWR                   ;WRITE.
5378 026372  026502                             RANWR                   ;WRITE.
5379 026374  026502                             RANWR                   ;WRITE.
5380 026376  026502                             RANWR                   ;WRITE.
5381 026400  026502                             RANWR                   ;WRITE.
5382 026402  026502                             RANWR                   ;WRITE.
5383 026404  026502                             RANWR                   ;WRITE.
5384 026406  026426                             RANRD                   ;READ.
5385 026410  026426                             RANRD                   ;READ.
5386 026412  026426                             RANRD                   ;READ.
5387 026414  026426                             RANRD                   ;READ.
5388 026416  026426                             RANRD                   ;READ.
5389 026420  026426                             RANRD                   ;READ.
5390 026422  026426                             RANRD                   ;READ.
5391 026424  026426                             RANRD                   ;READ.
5392
5393
5394
5395
5396
5397                                    ;       SUBROUTINE TO SET UP READ COMMANDS IN SEQUENCE TABLE.
5398                                    ;       INPUTS:
5399                                    ;       OUTPUTS:
5400                                    ;       REGISTERS:      R2
5401                                    ;       CALLS:
5402
5403 026426                             RANRD:  LET (R1)+ := #SRR               ;STORE SPACE RECORD REVERSE CMD
     026426  012721  104410                                                             MOV     #SRR,(R1)+
5404 026432                                     LET RANB := RANB + RANS
     026432  063737  003442  003440                                                     ADD     RANS,RANB
5405 026440                                     LET R2 := RANB CLR.BY #RNOPSC
     026440  013702  003440                                                             MOV     RANB,R2
     026444  042702  177740                                                             BIC     #RNOPSC,R2
5406 026450                                     LET (R1)+ := R2                 ;SET REPOSITION COUNT
     026450  010221                                                             MOV     R2,(R1)+
5407 026452                                     LET (R1)+ := #1                 ;DO ONCE
     026452  012721  000001                                                             MOV     #1,(R1)+
5408 026456                                     LET (R1)+ := #RANP              ;RANDOM PATTERN.
     026456  012721  000007                                                             MOV     #RANP,(R1)+
5409 026462                                     LET (R1)+ := #RDF               ;STORE READ FWD CMD.
     026462  012721  104001                                                             MOV     #RDF,(R1)+
5410 026466                                     LET (R1)+ := #DATCNT            ;SET BRF TO MAX TO READ RANDOM LENGTHS.
     026466  012721  010000                                                             MOV     #DATCNT,(R1)+
5411 026472                                     LET (R1)+ := R2                 ;SET RANDOM # OF OPERATIONS.
     026472  010221                                                             MOV     R2,(R1)+
5412 026474                                     LET (R1)+ := #RANP              ;RANDOM PATTERN.
     026474  012721  000007                                                             MOV     #RANP,(R1)+
5413 026500  000207                             RTS PC
```

```
                                ;       SUBROUTINE TO SET UP A WRITE COMMAND IN THE SEQUENCE TABLE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

                                RANWR:  LET (R1)+ := #WRT               ;STORE WRITE CMD.
026502                                                                          MOV     #WRT,(R1)+
026502  012721  104005
026506  004737  026526                  JSR PC,RANW                     ;STORE BRF, # OF OPERATIONS, PATTERN.
026512  000207                          RTS PC


                                ;       SUBROUTINE TO SET UP A WRITE/VERIFY COMMAND IN THE SEQUENCE TABLE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:
                                ;       CALLS:

026514                          RANWV:  LET (R1)+ := #WTV               ;STORE WRITE/VERIFY CMD.
026514  012721  104105                                                          MOV     #WTV,(R1)+
026520  004737  026526                  JSR PC,RANW                     ;STORE BRF, # OF OPERATIONS, PATTERN.
026524  000207                          RTS     PC


                                ;       SUBROUTINE TO STORE BRF, # OF OPERATIONS, PATTERN IN COMMAND
                                ;       SEQUENCE TABLE FOR WRITE AND WRITE/VERIFY COMMANDS.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2
                                ;       CALLS:

026526                          RANW:   LET (R1)+ := #DATCNT            ;SET BRF TO MAX FOR PATTERN GENERATION.
026526  012721  010000                                                          MOV     #DATCNT,(R1)+

                                        LET RANB := RANB + RANS         ;RANDOM BRF WILL BE GENERATED FOR EACH RECORD.
026532
026532  063737  003442  003440                                                  ADD     RANS,RANB

                                        LET R2 := RANB CLR.BY #RNOPSC
026540
026540  013702  003440                                                          MOV     RANB,R2
026544  042702  177740                                                          BIC     #RNOPSC,R2

                                        LET (R1)+ := R2                 ;SET RANDOM # OF OPERATIONS.
026550
026550  010221                                                                  MOV     R2,(R1)+

                                        LET (R1)+ := #RANP              ;RANDOM PATTERN.
026552
026552  012721  000007                                                          MOV     #RANP,(R1)+

                                        RTS PC                          ;RETURN.


                                        .EVEN


                                        ENDTST
026560                          L10035:
026560  104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL  TEST 3: WRITE AND READ STREAMING TEST.

                                ;++
                                ;
                                ;       THIS TEST STREAM WRITES 9000 RECORDS OF 4096 BYTES EACH.
                                ;       DATA IS THEN VERIFIED BY PERFORMING A REWIND OPERATION, FOLLOWED
                                ;       BY A READ FORWARD OPERATON FOR THE 9000 RECORDS.
                                ;
                                ;--

 026562                         BGNTST
 026562                         T3::

 026562                         IFB HERE EQ #0 THEN
 026562  105737  003340                                                TSTB    HERE
 026566  001012                                                        BNE     50501$
 026570                          PRINTF #TAPCAP
 026570  012746  022157                                                MOV     #TAPCAP,-(SP)
 026574  012746  000001                                                MOV     #1,-(SP)
 026600  010600                                                        MOV     SP,R0
 026602  104417                                                        TRAP    C$PNTF
 026604  062706  000004                                                ADD     #4,SP
 026610                          LET HERE :B= HERE + #1
 026610  105237  003340                                                INCB    HERE
 026614                         ENDIF
 026614                                                                50501$:
 026614                         LET RANDOM :B= #0               ;CLEAR THE RANDOM OPERATIONS FLAG.
 026614  105037  003533                                                CLRB    RANDOM
 026620                         LET EXPBOT :B= #0               ;CLEAR THE EXPECT BOT FLAG.
 026620  105037  003532                                                CLRB    EXPBOT
 026624  004737  006772         JSR PC, SETCH                   ;SET CHARACTERISTICS.
 026630  004737  007016         JSR PC, SETRW                   ;SET REWIND COMMAND IN BUFFER.
 026634                         LET STAFLG :B= #0               ;CLEAR THE START FLAG.
 026634  105037  003546                                                CLRB    STAFLG
 026640                         LET (R1) := #END                ;PLACE END FLAG IN SEQUENCE TABLE.
 026640  012711  177777                                                MOV     #END,(R1)
 026644  004737  007036         JSR PC, EXALL                   ;REWIND ALL UNITS.
 026650  004737  017666         JSR PC, FIRSTU                  ;FIND THE FIRST UNIT TO TEST (UUT)

                                ; ***************************************************************

                                ;WRITE AND READ EACH UNIT IN TURN BEFORE GOING ON TO THE NEXT.

                                ; ***************************************************************

 026654                         WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
 026654                                                                50502$:
 026654  026527  002536  177777                                        CMP     DEVTBL(R5),#END
 026662  001550                                                        BEQ     50503$

 026664                         LET BTPT := BTADDR(R5)          ;CLEAR BAD SPOT COUNTER
 026664  016537  002550  003526                                        MOV     BTADDR(R5),BTPT
 026672                         LET @BTPT := #0                 ;START FROM BOT
 026672  005077  154630                                                CLR     @BTPT
 026676                         LET STREAM :B= #255.            ;SET FLAG - WE'RE GOING TO STREAM
 026676  112737  000377  003544                                        MOVB    #255.,STREAM
 026704                          LET R1 := #CMDSEQ              ;SETUP SEQUENCE TABLE ADDRESS
```

```
026704  012701  003554                                                          MOV     #CMDSEQ,R1
026710                                  LET (R1)+ := #WRT           ;WRITE COMMAND
026710  012721  104005                                                          MOV     #WRT,(R1)+
026714                                  LET (R1)+ := #DATCNT  ;4096-BYTE RECORD LENGTH.
026714  012721  010000                                                          MOV     #DATCNT,(R1)+
026720                                  LET (R1)+ := #9000.   ;WRITE 9000 RECORDS.
026720  012721  021450                                                          MOV     #9000.,(R1)+
026724                                  LET (R1)+ := #5             ;GENERATE AND WRITE PATTERN 5.
026724  012721  000005                                                          MOV     #5,(R1)+
026730                                  LET (R1)  := #END           ;SET END OF SEQUENCE TABLE.
026730  012711  177777                                                          MOV     #END,(R1)
026734                                  LET R1 := #CMDSEQ           ;SEQ. TABLE ADDRESS FOR SUBR. 'SETUP'.
026734  012701  003554                                                          MOV     #CMDSEQ,R1
026740  004737  010000                  JSR PC, SETUP         ;SETUP THE COMMAND TABLE
026744                                  LET R5SAVE := R5            ;SAVE R5, EH?
026744  010537  003460                                                          MOV     R5,R5SAVE

026750                                    WHILE NCNT LT NCNT1 DO    ;WHILE MORE RECORDS SHOULD BE WRITTEN:
026750                                                                          50504$:
026750  023737  003420  003422                                                  CMP     NCNT,NCNT1
026756  002022                                                                  BGE     50505$

026760  004737  007672                     JSR PC, CMDAC            ;SAVE ASCII COMMAND FOR ERROR MESSAGE
026764  004737  010774                     JSR PC, EXCUTE           ;ISSUE COMMAND TO UNIT.
026770  004737  011470                     JSR PC, GOWAIT           ;GO WAIT FOR DONE TO SET
026774                                  IF DEVTBL(R5) EQ #NINUSE THEN
026774  026527  002536  177774                                                  CMP     DEVTBL(R5),#NINUSE
027002  001005                                                                  BNE     50506$
027004                                  LET NCNT := NCNT1 - #1
027004  013737  003422  003420                                                  MOV     NCNT1,NCNT
027012  005337  003420                                                          DEC     NCNT
027016                                  ENDIF
027016                                                                          50506$:
027016                                     LET NCNT := NCNT + #1    ;UPDATE THE RECORD COUNT
027016  005237  003420                                                          INC     NCNT
027022                                    ENDDO                     ;END OF RECORD 'DO' LOOP
027022  000752                                                                  BR      50504$
027024                                                                          50505$:
027024                                  LET R1 := #CMDSEQ           ;RESET R1 TO TOP OF TABLE
027024  012701  003554                                                          MOV     #CMDSEQ,R1
027030  004737  007016                  JSR     PC,SETRW            ;ISSUE REWIND
027034                                  LET (R1) := #END            ;PLACE END FLAG IN SEQUENCE TABLE
027034  012711  177777                                                          MOV     #END,(R1)
027040                                  LET -(SP) := L$LUN          ;SAVE THE CURRENT LUN
027040  013746  002074                                                          MOV     L$LUN,-(SP)
027044  004737  007036                  JSR     PC,EXALL                     ;DO REWIND, NOW
027050                                  LET R5 := R5SAVE            ;RESTORE R5
027050  013705  003460                                                          MOV     R5SAVE,R5
027054                                  LET L$LUN := (SP)+          ;RESTORE THE CURRENT LUN
027054  012637  002074                                                          MOV     (SP)+,L$LUN
027060                                  IF DEVTBL(R5) NE #NINUSE THEN
027060  026527  002536  177774                                                  CMP     DEVTBL(R5),#NINUSE
027066  001431                                                                  BEQ     50507$
027070                                   LET R1 := #CMDSEQ          ;TOP OF COMMAND TABLE
027070  012701  003554                                                          MOV     #CMDSEQ,R1
027074                                   LET (R1)+ := #RDF          ;READ FORWARD COMMAND
027074  012721  104001                                                          MOV     #RDF,(R1)+
```

```
5529 027100
     027100  012721  010000          LET (R1)+ := #DATCNT          ;4096 BYTE RECORDS
                                                    MOV     #DATCNT,(R1)+
5530 027104
     027104  012721  021450          LET (R1)+ := #9000.           ;9000 ITERATIONS
                                                    MOV     #9000.,(R1)+
5531 027110
     027110  012721  000005          LET (R1)+ := #5                       ;READ PATTERN NUMBER 5
                                                    MOV     #5,(R1)+
5532 027114
     027114  012721  177777          LET (R1)+ := #END             ;TABLE TERMINATOR
                                                    MOV     #END,(R1)+
5533 027120
     027120  012701  003554          LET R1 := #CMDSEQ             ;TOP OF TABLE, AGAIN!
                                                    MOV     #CMDSEQ,R1
5534 027124  004737  010000          JSR PC, SETUP                 ;SET UP THE COMMAND TABLE
5535 027130                          LET VFYFLG :B= #1             ;ALLOW THE DATA VERIFY
     027130  112737  000001  003534                                MOVB    #1,VFYFLG
5536 027136
     027136  010537  003460          LET R5SAVE := R5              ;SAVE R5
                                                    MOV     R5,R5SAVE
5537 027142  004737  016530          JSR PC, VFYDAT                ;GO OFF AND CHECK READ OPERATIONS
5538 027146                          LET R5 := R5SAVE              ;RESTORE R5
     027146  013705  003460                                       MOV     R5SAVE,R5
5539 027152                          ENDIF
     027152                                                       50507$:
5540 027152                          LET NCNT := #0                ;CLEAR RECORD COUNT
     027152  005037  003420                                       CLR     NCNT
5541 027156                          LET R3 := #ENDERF             ;END OF BUFFER ADDRESS
     027156  012703  003506                                       MOV     #ENDERF,R3
5542 027162  004737  012110          JSR PC, CLRERR                ;RESET THE ERROR BUFFER
5543 027166                          LET VFYFLG :B= #0             ;CLEAR VERIFY FLAG
     027166  105037  003534                                       CLRB    VFYFLG
5544 027172                          LET EXPBOT :B= #0             ;CLEAR EXPECT BOT FLAG.
     027172  105037  003532                                       CLRB    EXPBOT
5545 027176  004737  017734          JSR PC, NEXTU                 ;GET NEXT UNIT TO TEST (UUT).
5546
5547 027202                          ENDDO                         ;END OF UUT LOOP
     027202  000624                                               BR      50502$
     027204                                                       50503$:
5548
5549 027204                          LET STREAM :B= #0             ;CLEAR STREAMING FLAG FOR OTHER TESTS.
     027204  105037  003544                                       CLRB    STREAM
5550 027210                          LET ALLEOT :B= #0             ;RESET THE UNITS @ EOT STATUS
     027210  105037  003543                                       CLRB    ALLEOT
5551 027214                          LET RPTFLG :B= #1             ;REQUEST A REPORT
     027214  112737  000001  003535                                MOVB    #1,RPTFLG
5552 027222                          LET R1 := #CMDSEQ             ;TOP OF TABLE
     027222  012701  003554                                       MOV     #CMDSEQ,R1
5553 027226  004737  007016          JSR PC, SETRW                 ;STORE THE REWIND COMMAND
5554 027232                          LET (R1) := #END              ;TERMINATOR
     027232  012711  177777                                       MOV     #END,(R1)
5555 027236  004737  007036          JSR PC, EXALL                 ;REWIND AND REPORT STATUS FOR ALL UNITS
5556
5557 027242                          EXIT    TST                   ;EXIT TEST
     027242  104432                                               TRAP    C$EXIT
     027244  000002                                               .WORD   L10036-.
5558                                 .EVEN                         ;JUST IN CASE.
5559
5560 027246                          ENDTST
     027246                  L10036:
     027246  104401                                               TRAP    C$ETST
```

```
5562
5563                                    .SBTTL  TEST 4: WRITE COMPATABILITY/WRITE UTILITY.
5564
5565                                    ;++
5566                                    ; TEST TO WRITE RECORDS FROM BOT TO EOT.
5567                                    ;--
5568
5569 027250                                     BGNTST
     027250                             T4::
5570
5571 027250
     027250  112737  000001  003533             LET RANDOM :B= #1              ;SET THE RANDOM OPERATIONS FLAG.
                                                                                        MOVB    #1,RANDOM
5572 027256
     027256  105037  003532                     LET EXPBOT :B= #0              ;CLEAR EXPECT BOT FLAG.
                                                                                        CLRB    EXPBOT
5573 027262
     027262  012702  010000                     LET R2 := #DATCNT - #1         ;SET UP THE RECORD LENGTH MASK.
     027266  005302                                                                    MOV     #DATCNT,R2
                                                                                        DEC     R2
5574 027270
   . 027270  010237  003436                     LET LENMSK := COMP R2          ;ALLOW MAXIMUM BUFFER.
     027274  005137  003436                                                            MOV     R2,LENMSK
                                                                                        COM     LENMSK
5575 027300  004737  006772                     JSR PC,SETCH                   ;CMD 1 = SET CHARACTERISTIC.
5576 027304  004737  007016                     JSR PC,SETRW                   ;CMD2=REWIND
5577 027310                                     LET STAFLG :B= #0              ;CLEAR START FLAG
     027310  105037  003546                                                            CLRB    STAFLG
5578 027314                                     REPEAT                         ;REPEAT TO EOT.
     027314                                                                     50510$:
5579 027314                                       WHILE R1 LT #SEQEND DO       ;WHILE THERE IS MORE ROOM IN SEQ TABLE:
     027314                                                                     50511$:
     027314  020127  003644                                                            CMP     R1,#SEQEND
     027320  002003                                                                   BGE     50512$
5580 027322  004737  026502                       JSR PC,RANWR                   ;STORE A WRITE CMD IN SEQUENCE TABLE.
5581 027326                                       ENDDO
     027326  000772                                                                   BR      50511$
     027330                                                                     50512$:
5582 027330                                       LET (R1) := #END             ;STORE END OF SEQUENCE CODE IN TABLE.
     027330  012711  177777                                                            MOV     #END,(R1)
5583 027334  004737  007036                       JSR   PC,EXALL               ;EXECUTE ALL CMDS IN SEQ TBL ON UNITS.
5584 027340                                       LET R1 := #CMDSEQ            ;INIT SEQ TBL POINTER.
     027340  012701  003554                                                            MOV     #CMDSEQ,R1
5585 027344                                     UNTIL R2 NE #0                 ;REPEAT UNTIL EOT IS REACHED
     027344  005702                                                                    TST     R2
     027346  001762                                                                   BEQ     50510$
5586 027350                                     LET ALLEOT :B= ALLEOT + #1     ;SET ALL UNITS @ EOT FLAG
     027350  105237  003543                                                            INCB    ALLEOT
5587 027354  000240                             NOP
5588 027356  000240                             NOP
5589 027360  000240                             NOP
5590 027362  004737  030364                     JSR     PC,T5WEOT              ;WRITE ONE RECORD BEYOND EOT ON ALL UNITS
5591                                                                            ;SO THAT SHORTER READ STOP DISTANCE
5592                                                                            ;SHALL POSITION HEAD IN CLEAN IRG GAP
5593                                                                            ;READ REV THAT EXTRA REC TO RE-POSITION TAPE
5594 027366                                     LET ALLEOT :B= #0              ;CLEAR ALL UNITS @ EOT FLAG
     027366  105037  003543                                                            CLRB    ALLEOT
5595 027372  004737  007016                     JSR PC,SETRW                   ;STORE REWIND IN SEQ TBL.
5596 027376                                     LET (R1) := #END               ;STORE END IN SEQ TBL.
     027376  012711  177777                                                            MOV     #END,(R1)
5597 027402  004737  007036                     JSR PC,EXALL                   ;EXECUTE REWIND CMD ON ALL UNITS
```

```
 5598 027406                            EXIT    TST
      027406  104432                                                  TRAP    C$EXIT
      027410  000002                                                  .WORD   L10037-.
 5599                                   .EVEN
 5600
 5601 027412                            ENDTST
      027412                    L10037:
      027412  104401                                                  TRAP    C$ETST
 5602
 5603                                   .SBTTL  TEST 5:  READ COMPATABILITY/READ UTILITY.
 5604
 5605                                   ;++
 5606                                   ; TEST TO READ ENTIRE TAPE FORWARD AND REVERSE.
 5607                                   ;--
 5608 027414                            BGNTST
      027414                    T5::
 5609
 5610 027414                            LET RANDOM :B= #1               ;SET THE RANDOM OPERATIONS FLAG.
      027414  112737  000001  003533                                    MOVB    #1,RANDOM
 5611 027422                            LET EXPBOT :B= #1               ;SET EXPECT BOT FLAG.
      027422  112737  000001  003532                                    MOVB    #1,EXPBOT
 5612 027430  004737  006772            JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
 5613 027434  004737  007016            JSR PC,SETRW                    ;CMD2=REWIND.
 5614 027440                            LET STAFLG :B= #0               ;CLEAR START FLAG
      027440  105037  003546                                            CLRB    STAFLG
 5615 027444                            LET (R1)+ := #RDF               ;CMD3 = READ FORWARD.
      027444  012721  104001                                            MOV     #RDF,(R1)+
 5616 027450                            LET (R1)+ := #DATCNT            ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
      027450  012721  010000                                            MOV     #DATCNT,(R1)+
 5617 027454                            LET (R1)+ := #77777             ;SET RECORD COUNT TO MAX FOR WHOLE TAPE.
      027454  012721  077777                                            MOV     #77777,(R1)+
 5618 027460                            LET (R1)+ := #RANP              ;PATTERN = RANDOM.
      027460  012721  000007                                            MOV     #RANP,(R1)+
 5619 027464                            LET (R1) := #END                ;STORE END OF SEQUENCE CODE IN TABLE.
      027464  012711  177777                                            MOV     #END,(R1)
 5620 027470  004737  007036            JSR     PC,EXALL                ;EXECUTE ALL CMDS IN SEQ TBL ON ALL UNITS.
 5621                                ;  LET ALLEOT :B= ALLEOT + #1       ;FLAG TO ALLOW ALL UNITS AT EOT TO READ REV
 5622                                ;  LET R1 := #CMDSEQ                ;INIT CMD SEQ TBL POINTER.
 5623                                ;  LET (R1)+ := #RDR                 ;CMD1 = READ REVERSE.
 5624                                ;  LET (R1)+ := #DATCNT              ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
 5625                                ;  LET (R1)+ := #77777               ;RECORD COUNT = MAX FOR WHOLE TAPE.
 5626                                ;  LET (R1)+ := #RANP                ;PATTERN = RANDOM.
 5627                                ;  LET (R1) := #END                  ;STORE END OF SEQUENCE CODE IN TABLE.
 5628                                ;  JSR     PC,EXALL                ;GO EXECUTE READ REV. OF ENTIRE TAPE.
 5629 027474                            LET ALLEOT :B= #0               ;CLEAR ALL UNITS @ EOT FLAG
      027474  105037  003543                                            CLRB    ALLEOT
 5630 027500                            LET R1 := #CMDSEQ               ;TOP OF COMMAND TABLE
      027500  012701  003554                                            MOV     #CMDSEQ,R1
 5631 027504  004737  007016            JSR     PC,SETRW                ;ISSUE A REWIND COMMAND (TWR)
 5632 027510                            LET (R1)+ := #END               ;TERMINATOR
      027510  012721  177777                                            MOV     #END,(R1)+
 5633 027514  004737  007036            JSR     PC,EXALL                ;ISSUE THE REWIND
 5634
 5635 027520                            EXIT    TST
      027520  104432                                                    TRAP    C$EXIT
      027522  000002                                                    .WORD   L10040-.
 5636                                   .EVEN
```

```
   5637
   5638 027524                          ENDTST
        027524                  L10040:
        027524  104401                                TRAP    C$ETST
```

```
 5640
 5641                                   .SBTTL  TEST 6: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.
 5642
 5643                                   ;++
 5644                                   ; TEST TO EXECUTE  OPERATOR SELECTED COMMAND SEQUENCE.
 5645                                   ;--
 5646
 5647 027526                                   BGNTST
      027526                            T6::
 5648
 5649 027526                                   LET RANDOM :B= #0               ;CLEAR RAMDOM MODE FLAG.
      027526  105037  003533                                                          CLRB    RANDOM
 5650 027532                                   LET EXPBOT :B= #1               ;SET EXPECT BOT FLAG.
      027532  112737  000001  003532                                                  MOVB    #1,EXPBOT
 5651 027540                                   LET IRE :B= PIRE                ;MOVE INHIBIT RFC ERROR REPORT FLAG.
      027540  113737  002213  003540                                                  MOVB    PIRE,IRE
 5652 027546  004737  006772                   JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
 5653 027552                                   LET CMDSEQ+2 := CHAR            ;MOVE CHAR CODE FROM P TBL TO SEQ TBL.
      027552  013737  002220  003556                                                  MOV     CHAR,CMDSEQ+2
 5654 027560                                   LET R2 := #CMDD                 ;R2 POINTS TO CMD2 IN SOFT P TABLE.
      027560  012702  002222                                                          MOV     #CMDD,R2
 5655 027564  004737  030342                   JSR     PC,PTCMDS               ;MOVE CMD 2 FROM P TBL TO SEQ TBL.
 5656 027570  004737  030342                   JSR     PC,PTCMDS               ;MOVE CMD 3 FROM P TBL TO SEQ TBL.
 5657 027574  004737  030342                   JSR     PC,PTCMDS               ;MOVE CMD 4 FROM P TBL TO SEQ TBL.
 5658 027600  004737  030342                   JSR     PC,PTCMDS               ;MOVE CMD 5 FROM P TBL TO SEQ TBL.
 5659 027604  004737  030342                   JSR     PC,PTCMDS               ;MOVE CMD 6 FROM P TBL TO SEQ TBL.
 5660 027610  004737  030342                   JSR     PC,PTCMDS               ;MOVE CMD 7 FROM P TBL TO SEQ TBL.
 5661 027614  004737  030342                   JSR     PC,PTCMDS               ;MOVE END CMD FROM P TBL TO SEQ TBL.
 5662 027620                                   LET JLOOP := #0                 ;CLEAR JMP CMD LOOP COUNT.
      027620  005037  003450                                                          CLR     JLOOP
 5663 027624                                   LET STAFLG :B= #0               ;CLEAR START FLAG
      027624  105037  003546                                                          CLRB    STAFLG
 5664 027630                                   LET R1 := #CMDSEQ               ;INIT SEQUENCE TABLE POINTER.
      027630  012701  003554                                                          MOV     #CMDSEQ,R1
 5665
 5666          000000                   $BRJMP=0        ;ENABLE JMP SUBSTITUTION FOR BR, IF NECESSARY.
 5667
 5668 027634                            3$:    WHILE (R1) NE #END DO           ;WHILE THERE ARE CMDS LEFT IN SEQUENCE TBLE:
      027634                                                                          50513$:
      027634  021127  177777                                                          CMP     (R1),#END
      027640  001002                                                                  BNE     .+6
      027642  000137  030304                                                          JMP     50514$
 5669 027646  022711  000040                      CMP   #JMP.C,(R1)          ;IS THIS A JUMP CMD?
 5670 027652  001024                             BNE   6$                   ;BR IF NOT.
 5671 027654                                     LET R1 := R1 + #2          ;POINT TO BRF.
      027654  062701  000002                                                          ADD     #2,R1
 5672 027660  012137  003452                     MOV   (R1)+,JLOC           ;SAVE BRF (LOCATION).
 5673 027664  022137  003450                     CMP (R1)+,JLOOP            ;HAS LOOP COUNT BE SATISFIED?
 5674 027670  001003                             BNE   1$                   ;IF NOT, JMP AGAIN.
 5675 027672                                     LET R1 := R1 + #2          ;IF SO, ADJUST SEQ POUNTER
      027672  062701  000002                                                          ADD     #2,R1
 5676 027676  000756                             BR    3$                   ;AND GO TO NEXT COMMAND.
 5677 027700                            1$:      LET JLOOP := JLOOP + #1    ;UPDATE THE LOOP COUNT.
      027700  005237  003450                                                          INC     JLOOP
 5678 027704                                     LET R1 := #CMDSEQ          ;INIT CMD SEQ TABLE POINTER.
      027704  012701  003554                                                          MOV     #CMDSEQ,R1
 5679 027710  005337  003452            2$:      DEC   JLOC                 ;DECR LOCATION COUNTER.
```

```
 5680 027714  001747                        BEQ   3$                    ;IF THIS IS THE RIGHT LOCATION TO JMP TO, GO SET UP.
 5681 027716                                LET R1 := R1 + #10            ;IF NOT, UPDATE SEQ POINTER TO NEXT CMD.
      027716  062701  000010                                                    ADD     #10,R1
 5682 027722  000772                        BR    2$                      ;DO IT AGAIN.
 5683 027724  022711  000020          6$:   CMP   #DLY.C,(R1)             ;DELAY?
 5684 027730  001026                        BNE   4$                      ;BR IF NOT.
 5685 027732                                LET R1 := R1 + #4             ;R1 = LOCATION OF N COUNT.
      027732  062701  000004                                                    ADD     #4,R1
 5686 027736                                LET TIME2 := (R1)             ;SAVE N COUNT.
      027736  011137  003446                                                    MOV     (R1),TIME2
 5687 027742                          7$:   DELAY 10.                     ;GO TO SUPER-WAIT 1 MSEC.
      027742  012727  000012                                                    MOV     #10.,(PC)+
      027746  000000                                                           .WORD   0
      027750  013727  002116                                                    MOV     L$DLY,(PC)+
      027754  000000                                                           .WORD   0
      027756  005367  177772                                                    DEC     -6(PC)
      027762  001375                                                           BNE     .-4
      027764  005367  177756                                                    DEC     -22(PC)
      027770  001367                                                           BNE     .-20
 5688 027772  005337  003446                        DEC   TIME2
 5689 027776  001361                        BNE   7$
 5690 030000                                LET R1 := R1 + #4             ;POINT TO NEXT CMD.
      030000  062701  000004                                                    ADD     #4,R1
 5691 030004  000713                        BR    3$                      ;GO CHECK NEXT CMD.
 5692 030006  004737  010000          4$:   JSR   PC,SETUP                ;GO SETUP THE COMMAND BLOCK.
 5693 030012                                WHILE NCNT LT NCNT1 DO        ;WHILE THERE ARE RECORDS REMAINING:
      030012                                                                50515$:
      030012  023737  003420  003422                                           CMP     NCNT,NCNT1
      030020  002402                                                           BLT     .+6
      030022  000137  030274                                                    JMP     50516$
 5694 030026  004737  007672                JSR PC,CMDAC                  ;STORE CMD ASCII IN ERROR MSG.
 5695 030032  004737  007332                JSR PC,EXSUB                  ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATUS.
 5696 030036                                IF CMDWRD EQ #GES THEN        ;IF CMD IS GET STATUS THEN:
      030036  023727  003426  100017                                           CMP     CMDWRD,#GES
      030044  001402                                                           BEQ     .+6
      030046  000137  030056                                                    JMP     50517$
 5697 030052  004737  020212                  JSR PC,PRXST                ;PRINT EXTENDED STATUS REGISTERS.
 5698 030056                                ENDIF
      030056                                                                50517$:
 5699 030056  004737  020274                JSR PC,CKHAE                  ;CHECK HALT AFTER EACH CMD FLAG.
 5700 030062                                LET R2 := #1                  ;SET ALL UNITS AT BOT/EOT.
      030062  012702  000001                                                    MOV     #1,R2
 5701 030066  004737  017666                JSR PC,FIRSTU                 ;FIND FIRST UNIT.
 5702 030072                                WHILE DEVTBL(R5) NE #END DO   ;WHILE THERE ARE MORE UNITS:
      030072                                                                50520$:
      030072  026527  002536  177777                                           CMP     DEVTBL(R5),#END
      030100  001002                                                           BNE     .+6
      030102  000137  030200                                                    JMP     50521$
 5703 030106                                  IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
      030106  032737  000400  003426                                           BIT     #MOD.CO,CMDWRD
      030114  001002                                                           BNE     .+6
      030116  000137  030144                                                    JMP     50522$
 5704 030122                                    IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT THEN:
      030122  032765  000002  003516                                           BIT     #XO.BOT,EOTFLG(R5)
      030130  001402                                                           BEQ     .+6
      030132  000137  030140                                                    JMP     50523$
 5705 030136                                      LET R2 := #0                ;CLEAR EOT/BOT FLAG.
```

```
     030136  005002                                                                 CLR     R2
     030140                                 ENDIF
     030140                                                                         50523$:
     030140                               ELSE                     ;ELSE IF CMD IS NOT REVERSE:
     030140  000137  030172                                                         JMP     50524$
     030144                                                                         50522$:
     030144                                 IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
     030144  032765  000001  003516                                                 BIT     #XO.EOT,EOTFLG(R5)
     030152  001406                                                                 BEQ     50525$
     030154  032737  000001  003426                                                 BIT     #CMD.CO,CMDWRD
     030162  001402                                                                 BEQ     .+6
     030164  000137  030172                                                         JMP     50526$
     030170                                                                         50525$:
                                                                   ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
     030170                                   LET R2 := #0           ;CLEAR EOT/BOT FLAG.
     030170  005002                                                                 CLR     R2
     030172                                 ENDIF
     030172                                                                         50526$:
     030172                               ENDIF
     030172                                                                         50524$:
     030172  004737  017734               JSR PC,NEXTU             ;FIND NEXT UNIT
     030176                             ENDDO                      ;
     030176  000735                                                                 BR      50520$
     030200                                                                         50521$:
     030200                             IF R2 EQ #1 THEN           ;IF ALL UNIT ARE AT EOT/BOT THEN:
     030200  020227  000001                                                         CMP     R2,#1
     030204  001402                                                                 BEQ     .+6
     030206  000137  030254                                                         JMP     50527$
     030212                               LET NCNT1 := NCNT + #1   ;FORCE TERMINATION OF COMMAND.
     030212  013737  003420  003422                                                 MOV     NCNT,NCNT1
     030220  005237  003422                                                         INC     NCNT1
     030224                               LET ALLEOT :B= ALLEOT + #1 ;FLAG ALL UNITS AT EOT/BOT TO ALLOW VERIFY OF DATA
     030224  105237  003543                                                         INCB    ALLEOT
     030230                               IF CMDLG EQ #2 THEN      ;WHEN WRITING IS CURRENT COMMAND
     030230  023727  003434  000002                                                 CMP     CMDLG,#2
     030236  001402                                                                 BEQ     .+6
     030240  000137  030250                                                         JMP     50530$
     030244  004737  030364                 JSR PC,T5WEOT          ;GO WRITE/READ REV ONE RECORD BEYOND EOT
     030250                               ENDIF
     030250                                                                         50530$:
     030250                             ELSE
     030250  000137  030260                                                         JMP     50531$
     030254                                                                         50527$:
     030254                               LET ALLEOT :B= #0 ;WHEN NOT ALL @EOT, CLEAR FLAG
     030254  105037  003543                                                         CLRB    ALLEOT
     030260                             ENDIF
     030260                                                                         50531$:
     030260                             LET NCNT := NCNT + #1      ;UPDATE RECORD COUNT.
     030260  005237  003420                                                         INC     NCNT
     030264                             LET PCMDWD := CMDWRD       ;SAVE PREVIOUS COMMAND WORD.
     030264  013737  003426  003432                                                 MOV     CMDWRD,PCMDWD
     030272                           ENDDO
     030272  000647                                                                 BR      50515$
     030274                                                                         50516$:
     030274  004737  016530           JSR    PC,VFYDAT             ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
                                                                   ;VERIFY THE LAST N RECORDS OF DATA.
```

```
 5730 030300                                ENDDO
      030300  000137  027634                                                  JMP     50513$
      030304                                                                  50514$:
 5731
 5732          177777                $BRJMP=-1       ;TURN OFF JMP SUBSTITUTION (SPMACJ CONTROL, ONLY).
 5733 030304                                LET ALLEOT :B= #0                ;NOT AT EOT
      030304  105037  003543                                                  CLRB    ALLEOT
 5734 030310                                LET RPTFLG :B= #1                ;REQUEST A REPORT
      030310  112737  000001  003535                                         MOVB    #1,RPTFLG
 5735 030316                                LET R1 := #CMDSEQ                ;SET UP THE COMMAND TABLE ADDRESS
      030316  012701  003554                                                  MOV     #CMDSEQ,R1
 5736 030322  004737  007016                JSR PC,SETRW                     ;STORE REWIND COMMAND
 5737 030326                                LET (R1) := #END                 ;TERMINATE THE TABLE
      030326  012711  177777                                                  MOV     #END,(R1)
 5738 030332  004737  007036                JSR PC,EXALL                     ;ISSUE THE REWINDS
 5739
 5740
 5741 030336                                EXIT    TST
      030336  104432                                                         TRAP    C$EXIT
      030340  000130                                                         .WORD   L10041-.
 5742
```

```
5744
5745                            ;       SUBROUTINE TO MOVE A COMMAND FROM THE SOFTWARE P TABLE TO
5746                            ;       THE COMMAND SEQUENCE TABLE.
5747                            ;       INPUTS:         R2 = POINTER TO SOFT "P" TABLE
5748                            ;       OUTPUTS:
5749                            ;       REGISTERS:      R3.
5750                            ;       CALLS:
5751
5752 030342                    PTCMDS: LET R3 := (R2)+ - #1 SHIFT +1   ;R3 = COMMAND TABLE INDEX.
        030342  012203                                                          MOV     (R2)+,R3
        030344  005303                                                          DEC     R3
        030346  006303                                                          ASL     R3
5753 030350                             LET (R1)+ := CMDTBL(R3)         ;MOVE COMMAND WORD.
        030350  016321  003656                                                  MOV     CMDTBL(R3),(R1)+
5754 030354                             LET (R1)+ := (R2)+              ;MOVE # OF BYTES.
        030354  012221                                                          MOV     (R2)+,(R1)+
5755 030356                             LET (R1)+ := (R2)+              ;MOVE # OF OPERATIONS.
        030356  012221                                                          MOV     (R2)+,(R1)+
5756 030360                             LET (R1)+ := (R2)+              ;MOVE PATTERN CODE.
        030360  012221                                                          MOV     (R2)+,(R1)+
5757 030362  000207                     RTS PC
5758
5759                            ;       SUBROUTINE TO WRITE THEN READ REVERSE ONE RECORD BEYOND EOT
5760                            ;       INPUTS:
5761                            ;       OUTPUTS:
5762                            ;       REGISTERS:
5763                            ;       CALLS:          CMDAC,EXSUB,CKHAE
5764
5765 030364  000240             T5WEOT: NOP
5766 030366  000240                     NOP
5767 030370  004737  007332             JSR PC,EXSUB                    ;WRITE ONE RECORD BEYOND EOT
5768 030374  004737  020274             JSR PC,CKHAE                    ;SO THAT READ SHORTER STOP DISTANCE
5769                                                                    ;SHALL POSITION HEAD IN CLEAN IRG GAP
5770 030400                             LET PCMDWD := CMDWRD            ;REPOSITION TAPE
        030400  013737  003426  003432                                           MOV     CMDWRD,PCMDWD
5771 030406                             LET CMDWRD := #RDR              ;BEFORE EXTRA RECORD
        030406  012737  104401  003426                                           MOV     #RDR,CMDWRD
5772 030414                             LET CMDLG := #4                 ;BY READING REVERSE
        030414  012737  000004  003434                                           MOV     #4,CMDLG
5773 030422                             LET CMDPKT := CMDWRD CLR.BY #BRF.C
        030422  013737  003426  002314                                           MOV     CMDWRD,CMDPKT
        030430  042737  004000  002314                                           BIC     #BRF.C,CMDPKT
5774 030436                             LET CMDSAV := CMDPKT            ;THAT RECORD TO ALLOW
        030436  013737  002314  003430                                           MOV     CMDPKT,CMDSAV
5775 030444                             LET CMDPKT+CP.ADL := DATARD     ;NEXT COMMAND IN THE
        030444  013737  003416  002316                                           MOV     DATARD,CMDPKT+CP.ADL
5776 030452  004737  007672             JSR PC,CMDAC                    ;TABLE TO BE EXECUTED
5777 030456  004737  007332             JSR PC,EXSUB
5778 030462  004737  020274             JSR PC,CKHAE
5779 030466  000207                     RTS PC
5780                                    .EVEN
5781 030470                             ENDTST
        030470                  L10041:
        030470  104401                                                          TRAP    C$ETST
5782 030472                             ENDMOD
```

```
                                .TITLE PARAMETER CODING
                                .SBTTL  HARDWARE PARAMETER CODING SECTION
030472                                  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.
                                ;--
030472                                  BGNHRD
030472  000024                                                          .WORD L10042-L$HARD/2
030474                          L$HARD::
030474                                  GPRMA   TS4ADR,0,0,160002,177564,YES
030474  000031                                                          .WORD   T$CODE
030476  030520                                                          .WORD   TS4ADR
030500  160002                                                          .WORD   T$LOLIM
030502  177564                                                          .WORD   T$HILIM
030504                                  GPRMD   TS4VCT,2,0,777,60,776,YES
030504  001032                                                          .WORD   T$CODE
030506  030535                                                          .WORD   TS4VCT
030510  000777                                                          .WORD   777
030512  000060                                                          .WORD   T$LOLIM
030514  000776                                                          .WORD   T$HILIM
030516                                  EXIT HRD
030516  013004                                                          .WORD   T$CODE
                                .NLIST  BEX
030520  124 123 123             TS4ADR: .ASCIZ  /TSSR ADDRESS/
030535  126 105 103             TS4VCT: .ASCIZ  /VECTOR/
                                .LIST   BEX
                                .EVEN
030544                                  ENDHRD
030544                          L10042:                                 .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.
                                ;--

030544                                  BGNSFT
030544  000531
030546                          L$SOFT::                                .WORD L10043-L$SOFT/2

030546                                  GPRML   CLRM,0,1,YES
030546  000130                                                          .WORD   T$CODE
030550  031324                                                          .WORD   CLRM
030552  000001                                                          .WORD   1
030554                                  GPRML   RRVM,0,400,YES
030554  000130                                                          .WORD   T$CODE
030556  031343                                                          .WORD   RRVM
030560  000400                                                          .WORD   400
030562                                  GPRML   HAEM,2,1,YES
030562  001130                                                          .WORD   T$CODE
030564  031372                                                          .WORD   HAEM
030566  000001                                                          .WORD   1
030570                                  GPRML   RCVERM,2,400,YES
030570  001130                                                          .WORD   T$CODE
030572  031416                                                          .WORD   RCVERM
030574  000400                                                          .WORD   400
030576                                  GPRML   IRECM,4,1,YES
030576  002130                                                          .WORD   T$CODE
030600  031440                                                          .WORD   IRECM
030602  000001                                                          .WORD   1
030604                                  XFERT   NEXTSP
030604  004024                                                          .WORD   T$CODE
030606                                  GPRML   BADTM,4,400,YES
030606  002130                                                          .WORD   T$CODE
030610  031461                                                          .WORD   BADTM
030612  000400                                                          .WORD   400
030614                          NEXTSP: GPRML   DINTM,6,1,YES
030614  003130                                                          .WORD   T$CODE
030616  031506                                                          .WORD   DINTM
030620  000001                                                          .WORD   1
030622                                  GPRML   IREM,6,400,YES
030622  003130                                                          .WORD   T$CODE
030624  031531                                                          .WORD   IREM
030626  000400                                                          .WORD   400
030630                                  GPRML   RAMM,10,1,YES
030630  004130                                                          .WORD   T$CODE
030632  031564                                                          .WORD   RAMM
030634  000001                                                          .WORD   1

030636                                  GPRML   EWPT,12,1,YES
030636  005130                                                          .WORD   T$CODE
```

```
        030640  031610                                                  .WORD   EWPT
        030642  000001                                                  .WORD   1
5877    030644                  GPRML   CHGM,12,400,YES
        030644  005130                                                  .WORD   T$CODE
        030646  031646                                                  .WORD   CHGM
        030650  000400                                                  .WORD   400
5878    030652                  XFERF   ENDSP1
        030652  127044                                                  .WORD   T$CODE
5879    030654                  GPRMD   CHARM,14,0,377,0,777,YES
        030654  006032                                                  .WORD   T$CODE
        030656  031665                                                  .WORD   CHARM
        030660  000377                                                  .WORD   377
        030662  000000                                                  .WORD   T$LOLIM
        030664  000777                                                  .WORD   T$HILIM
5880    030666                  GPRMD   CMD2M,16,D,37,1,33,YES
        030666  007052                                                  .WORD   T$CODE
        030670  031712                                                  .WORD   CMD2M
        030672  000037                                                  .WORD   37
        030674  000001                                                  .WORD   T$LOLIM
        030676  000033                                                  .WORD   T$HILIM
5881    030700                  GPRMD   BPCRM,20,D,-1,1,DATCNT,YES
        030700  010052                                                  .WORD   T$CODE
        030702  031720                                                  .WORD   BPCRM
        030704  177777                                                  .WORD   -1
        030706  000001                                                  .WORD   T$LOLIM
        030710  010000                                                  .WORD   T$HILIM
5882    030712                  GPRMD   NUMBM,22,D,-1,1,77777,YES
        030712  011052                                                  .WORD   T$CODE
        030714  031732                                                  .WORD   NUMBM
        030716  177777                                                  .WORD   -1
        030720  000001                                                  .WORD   T$LOLIM
        030722  077777                                                  .WORD   T$HILIM
5883    030724                  GPRMD   PATTM,24,D,17,0,10,YES
        030724  012052                                                  .WORD   T$CODE
        030726  031752                                                  .WORD   PATTM
        030730  000017                                                  .WORD   17
        030732  000000                                                  .WORD   T$LOLIM
        030734  000010                                                  .WORD   T$HILIM
5884    030736                  GPRMD   CMD3M,26,D,37,1,33,YES
        030736  013052                                                  .WORD   T$CODE
        030740  031764                                                  .WORD   CMD3M
        030742  000037                                                  .WORD   37
        030744  000001                                                  .WORD   T$LOLIM
        030746  000033                                                  .WORD   T$HILIM
5885    030750                  GPRMD   BPCRM,30,D,-1,1,DATCNT,YES
        030750  014052                                                  .WORD   T$CODE
        030752  031720                                                  .WORD   BPCRM
        030754  177777                                                  .WORD   -1
        030756  000001                                                  .WORD   T$LOLIM
        030760  010000                                                  .WORD   T$HILIM
5886    030762                  GPRMD   NUMBM,32,D,-1,1,77777,YES
        030762  015052                                                  .WORD   T$CODE
        030764  031732                                                  .WORD   NUMBM
        030766  177777                                                  .WORD   -1
        030770  000001                                                  .WORD   T$LOLIM
        030772  077777                                                  .WORD   T$HILIM
5887    030774                  GPRMD   PATTM,34,D,17,0,10,YES
```

```
        030774  016052                                                  .WORD   T$CODE
        030776  031752                                                  .WORD   PATTM
        031000  000017                                                  .WORD   17
        031002  000000                                                  .WORD   T$LOLIM
        031004  000010                                                  .WORD   T$HILIM
5888    031006                          GPRMD   CMD4M,36,D,37,1,33,YES
        031006  017052                                                  .WORD   T$CODE
        031010  031772                                                  .WORD   CMD4M
        031012  000037                                                  .WORD   37
        031014  000001                                                  .WORD   T$LOLIM
        031016  000033                                                  .WORD   T$HILIM
5889    031020                          GPRMD   BPCRM,40,D,-1,1,DATCNT,YES
        031020  020052                                                  .WORD   T$CODE
        031022  031720                                                  .WORD   BPCRM
        031024  177777                                                  .WORD   -1
        031026  000001                                                  .WORD   T$LOLIM
        031030  010000                                                  .WORD   T$HILIM
5890    031032                          GPRMD   NUMBM,42,D,-1,1,77777,YES
        031032  021052                                                  .WORD   T$CODE
        031034  031732                                                  .WORD   NUMBM
        031036  177777                                                  .WORD   -1
        031040  000001                                                  .WORD   T$LOLIM
        031042  077777                                                  .WORD   T$HILIM
5891    031044                          GPRMD   PATTM,44,D,17,0,10,YES
        031044  022052                                                  .WORD   T$CODE
        031046  031752                                                  .WORD   PATTM
        031050  000017                                                  .WORD   17
        031052  000000                                                  .WORD   T$LOLIM
        031054  000010                                                  .WORD   T$HILIM
5892    031056                          GPRMD   CMD5M,46,D,37,1,33,YES
        031056  023052                                                  .WORD   T$CODE
        031060  032000                                                  .WORD   CMD5M
        031062  000037                                                  .WORD   37
        031064  000001                                                  .WORD   T$LOLIM
        031066  000033                                                  .WORD   T$HILIM
5893    031070                          GPRMD   BPCRM,50,D,-1,1,DATCNT,YES
        031070  024052                                                  .WORD   T$CODE
        031072  031720                                                  .WORD   BPCRM
        031074  177777                                                  .WORD   -1
        031076  000001                                                  .WORD   T$LOLIM
        031100  010000                                                  .WORD   T$HILIM
5894    031102                          GPRMD   NUMBM,52,D,-1,1,77777,YES
        031102  025052                                                  .WORD   T$CODE
        031104  031732                                                  .WORD   NUMBM
        031106  177777                                                  .WORD   -1
        031110  000001                                                  .WORD   T$LOLIM
        031112  077777                                                  .WORD   T$HILIM
5895    031114                          GPRMD   PATTM,54,D,17,0,10,YES
        031114  026052                                                  .WORD   T$CODE
        031116  031752                                                  .WORD   PATTM
        031120  000017                                                  .WORD   17
        031122  000000                                                  .WORD   T$LOLIM
        031124  000010                                                  .WORD   T$HILIM
5896    031126                          XFER    ENDSP2
        031126  002004                                                  .WORD   T$CODE
5897    031130                  ENDSP1: XFER    ENDSP
        031130  075004                                                  .WORD   T$CODE
```

```
5898 031132                          ENDSP2: GPRMD   CMD6M,56,D,37,1,33,YES
     031132  027052                                                  .WORD   T$CODE
     031134  032006                                                  .WORD   CMD6M
     031136  000037                                                  .WORD   37
     031140  000001                                                  .WORD   T$LOLIM
     031142  000033                                                  .WORD   T$HILIM
5899 031144                                  GPRMD   BPCRM,60,D,-1,1,DATCNT,YES
     031144  030052                                                  .WORD   T$CODE
     031146  031720                                                  .WORD   BPCRM
     031150  177777                                                  .WORD   -1
     031152  000001                                                  .WORD   T$LOLIM
     031154  010000                                                  .WORD   T$HILIM
5900 031156                                  GPRMD   NUMBM,62,D,-1,1,77777,YES
     031156  031052                                                  .WORD   T$CODE
     031160  031732                                                  .WORD   NUMBM
     031162  177777                                                  .WORD   -1
     031164  000001                                                  .WORD   T$LOLIM
     031166  077777                                                  .WORD   T$HILIM
5901 031170                                  GPRMD   PATTM,64,D,17,0,10,YES
     031170  032052                                                  .WORD   T$CODE
     031172  031752                                                  .WORD   PATTM
     031174  000017                                                  .WORD   17
     031176  000000                                                  .WORD   T$LOLIM
     031200  000010                                                  .WORD   T$HILIM
5902 031202                                  GPRMD   CMD7M,66,D,37,1,33,YES
     031202  033052                                                  .WORD   T$CODE
     031204  032014                                                  .WORD   CMD7M
     031206  000037                                                  .WORD   37
     031210  000001                                                  .WORD   T$LOLIM
     031212  000033                                                  .WORD   T$HILIM
5903 031214                                  GPRMD   BPCRM,70,D,-1,1,DATCNT,YES
     031214  034052                                                  .WORD   T$CODE
     031216  031720                                                  .WORD   BPCRM
     031220  177777                                                  .WORD   -1
     031222  000001                                                  .WORD   T$LOLIM
     031224  010000                                                  .WORD   T$HILIM
5904 031226                                  GPRMD   NUMBM,72,D,-1,1,77777,YES
     031226  035052                                                  .WORD   T$CODE
     031230  031732                                                  .WORD   NUMBM
     031232  177777                                                  .WORD   -1
     031234  000001                                                  .WORD   T$LOLIM
     031236  077777                                                  .WORD   T$HILIM
5905 031240                                  GPRMD   PATTM,74,D,17,0,10,YES
     031240  036052                                                  .WORD   T$CODE
     031242  031752                                                  .WORD   PATTM
     031244  000017                                                  .WORD   17
     031246  000000                                                  .WORD   T$LOLIM
     031250  000010                                                  .WORD   T$HILIM
5906 031252                                  GPRMD   CMD8M,76,D,37,1,33,YES
     031252  037052                                                  .WORD   T$CODE
     031254  032022                                                  .WORD   CMD8M
     031256  000037                                                  .WORD   37
     031260  000001                                                  .WORD   T$LOLIM
     031262  000033                                                  .WORD   T$HILIM
5907 031264                                  GPRMD   BPCRM,100,D,-1,1,DATCNT,YES
     031264  040052                                                  .WORD   T$CODE
     031266  031720                                                  .WORD   BPCRM
```

```
        031270  177777                                                  .WORD   -1
        031272  000001                                                  .WORD   T$LOLIM
        031274  010000                                                  .WORD   T$HILIM
5908    031276                          GPRMD   NUMBM,102,D,-1,1,77777,YES
        031276  041052                                                  .WORD   T$CODE
        031300  031732                                                  .WORD   NUMBM
        031302  177777                                                  .WORD   -1
        031304  000001                                                  .WORD   T$LOLIM
        031306  077777                                                  .WORD   T$HILIM
5909    031310                          GPRMD   PATTM,104,D,17,0,10,YES
        031310  042052                                                  .WORD   T$CODE
        031312  031752                                                  .WORD   PATTM
        031314  000017                                                  .WORD   17
        031316  000000                                                  .WORD   T$LOLIM
        031320  000010                                                  .WORD   T$HILIM
5910    031322                  ENDSP:
5911    031322                          XFER    JMPMSG
        C31322  120004                                                  .WORD   T$CODE
```

```
 5913
 5920                                          .NLIST  BEX
 5921 031324     103     114     105  CLRM:   .ASCIZ  /CLEAR COUNTERS/
 5922 031343     122     105     123  RRVM:   .ASCIZ  /RESET RANDOM VARIABLES/
 5923 031372     110     101     114  HAEM:   .ASCIZ  /HALT AFTER EACH CMD/
 5924 031416     120     122     111  RCVERM: .ASCIZ  /PRINT SOFT ERRORS/
 5925 031440     111     116     110  IRECM:  .ASCIZ  /INHIBIT RECOVERY/
 5926 031461     102     101     104  BADTM:  .ASCIZ  /BAD TAPE SPOT DETECT/
 5927 031506     104     111     123  DINTM:  .ASCIZ  /DISABLE INTERRUPTS/
 5928 031531     111     116     110  IREM:   .ASCIZ  /INHIBIT RFC ERROR REPORT/
 5929                                          .LIST   BEX
 5930                                          .EVEN
 5931 031562                          JMPMSG:
 5932 031562                                  XFER    JMPMS2
      031562  100004                                                .WORD   T$CODE
 5933 031564     103     117     116  RAMM:   .ASCIZ  /CONTROLLER RAM DUMP/
      031567     124     122     117
      031572     114     114     105
      031575     122     040     122
      031600     101     115     040
      031603     104     125     115
      031606     120     000
 5934 031610     105     116     101  EWPT:   .ASCIZ  /ENABLE EARLY WARNING MESSAGES/
      031613     102     114     105
      031616     040     105     101
      031621     122     114     131
      031624     040     127     101
      031627     122     116     111
      031632     116     107     040
      031635     115     105     123
      031640     123     101     107
      031643     105     123     000
 5935 031646     103     110     101  CHGM:   .ASCIZ  /CHANGE CMD SEQ/
      031651     116     107     105
      031654     040     103     115
      031657     104     040     123
      031662     105     121     000
 5936 031665     103     110     101  CHARM:  .ASCIZ  /CHARACTERISTICS CODE/
      031670     122     101     103
      031673     124     105     122
      031676     111     123     124
      031701     111     103     123
      031704     040     103     117
      031707     104     105     000
 5937 031712     103     115     104  CMD2M:  .ASCIZ  "CMD/2"
      031715     057     062     000
 5938 031720     102     122     106  BPCRM:  .ASCIZ  /BRF COUNT/
      031723     040     103     117
      031726     125     116     124
      031731     000
 5939 031732     043     040     117  NUMBM:  .ASCIZ  /# OF OPERATIONS/
      031735     106     040     117
      031740     120     105     122
      031743     101     124     111
      031746     117     116     123
      031751     000
 5940 031752     120     101     124  PATTM:  .ASCIZ  /PATTERN/
```

```
     031755     124     105     122
     031760     116     000
5941                                        .LIST   BEX
5942                                        .EVEN
5943 031762                           JMPMS2:
5944 031762                                 EXIT SFT
     031762  023004                                                  .WORD   T$CODE
5945
5946                                        .NLIST  BEX
5947 031764     103     115     104  CMD3M:  .ASCIZ  "CMD/3"
5948 031772     103     115     104  CMD4M:  .ASCIZ  "CMD/4"
5949 032000     103     115     104  CMD5M:  .ASCIZ  "CMD/5"
5950 032006     103     115     104  CMD6M:  .ASCIZ  "CMD/6"
5951 032014     103     115     104  CMD7M:  .ASCIZ  "CMD/7"
5952 032022     103     115     104  CMD8M:  .ASCIZ  "CMD/8"
5953                                        .LIST   BEX
5954 032030                                 ENDSFT
     032030                           L10043:                           .EVEN
5955
5956                                 ;*****************************************************************************
5957                                 ;*****************************************************************************
5958                                 ;       PATCH AREA
5959
5960 032030                           PATCH:: .BLKW   64.
5961                                 ;*****************************************************************************
5962                                 ;*****************************************************************************
5963
5964 032230                                 LASTAD
     032230  032244                                                  .EVEN
     032232  000004                                                  .WORD T$FREE
     032234                                                          .WORD T$SIZE
                                     L$LAST::
5965 032234                                 ENDMOD
```

```
 5967                                .SBTTL  HARD CODED P-TBL
 5968
 5969                                ;++
 5970                                ;DIAG IS PRE-PARAMETERIZED PER TBL
 5971                                ;--
 5972
 5973 032234                                BGNSETUP 1
 5974 032234                                  BGNPTAB
      032234  000000                                                    .WORD   0
      032236  000002                                                    .WORD   L10046-./2-1
      032240                        L10044:
 5975 032240  172522                            172522
 5976 032242  000224                            224
 5977 032244                                  ENDPTAB
      032244                        L10046:
 5978 032244                                ENDSETUP
 5979
 5980         000001                .END
```

K14


```
ACK.C = 100000 G
ADR   = 000020 G
ALLEOT  003543 G
ASSEMB= 000010
ATTNM   004431 G
AUDRPM  004741 G
AUTODM  024150
BADTM   031461
BADTSW  002211 G
BFSEQ   025332
BFSEQ0  025356
BFSEQ1  025430
BFSEQ2  025442
BFSEQ3  025534
BFSEQ4  025606
BFSEQ5  025650
BFSEQ6  025722
BFSEQ7  025754
BFSEQ8  026006
BFSEQ9  026040
BFSE10  026062
BGNFLG= 003474
BINC    016310
BIT0  = 000001 G
BIT00 = 000001 G
BIT01 = 000002 G
BIT02 = 000004 G
BIT03 = 000010 G
BIT04 = 000020 G
BIT05 = 000040 G
BIT06 = 000100 G
BIT07 = 000200 G
BIT08 = 000400 G
BIT09 = 001000 G
BIT1  = 000002 G
BIT10 = 002000 G
BIT11 = 004000 G
BIT12 = 010000 G
BIT13 = 020000 G
BIT14 = 040000 G
BIT15 = 100000 G
BIT2  = 000004 G
BIT3  = 000010 G
BIT4  = 000020 G
BIT5  = 000040 G
BIT6  = 000100 G
BIT7  = 000200 G
BIT8  = 000400 G
BIT9  = 001000 G
BOE   = 000400 G
BORERS  015534 G
BPCRM   031720
BRCPK   002330 G
BRFCNT  003424 G
BRF.C = 004000 G
BTADDR  002550 G
BTMSG1  015322
BTMSG2  015407
BTMSG3  015464
BTPT    003526 G
BTRPT   021034
BT0     003000 G
BT1     003052 G
BT2     003124 G
BT3     003176 G
BYTECO= 000403
CHAR    002220 G
CHARM   031665
CHGFLG  002217 G
CHGM    031646
CHKERR  012456 G
CH.EAI= 000040 G
CH.ERI= 000020 G
CH.ESS= 000200 G
CKDATA  017252 G
CKDCNT  017662
CKDFF   017664
CKHAE   020274 G
CKHRTN  020362
CLKSVC  010746 G
CLN   = 101012 G
CLRERR  012110 G
CLRFLG  002204 G
CLRM    031324
CMDAC   007672 G
CMDASC  003744 G
CMDD    002222 G
CMDLG   003434 G
CMDPKM  004176 G
CMDPKT  002314 G
CMDSAV  003430 G
CMDSEQ  003554 G
CMDSE2  003564 G
CMDTBL  003656 G
CMDWRD  003426 G
CMD.C0= 000001 G
CMD.C1= 000002 G
CMD.C2= 000004 G
CMD.C3= 000010 G
CMD.C4= 000020 G
CMD2M   031712
CMD3M   031764
CMD4M   031772
CMD5M   032000
CMD6M   032006
CMD7M   032014
CMD8M   032022
CMPDAT  003410 G
CNTBGN= 002560
CNTEND= 003330
CNTLEN= 000550 G
CODELM  004066 G
COUNTE= 050436
CP.ADH= 000004 G
CP.ADL= 000002 G
CP.CMD= 000000 G
CP.CNT= 000006 G
CRLF    005307 G
CRLFSP  005312 G
CTCC    003456 G
CVC.C = 040000 G
C$AU  = 000052
C$AUTO= 000061
C$BRK = 000022
C$BSEG= 000004
C$BSUB= 000002
C$CEFG= 000045
C$CLCK= 000062
C$CLEA= 000012
C$CLOS= 000035
C$CLP1= 000006
C$CVEC= 000036
C$DCLN= 000044
C$DODU= 000051
C$DRPT= 000024
C$DU  = 000053
C$EDIT= 000003
C$ERDF= 000055
C$ERHR= 000056
C$ERRO= 000060
C$ERSF= 000054
C$ERSO= 000057
C$ESCA= 000010
C$ESEG= 000005
C$ESUB= 000003
C$ETST= 000001
C$EXIT= 000032
C$GETB= 000026
C$GETW= 000027
C$GMAN= 000043
C$GPHR= 000042
C$GPLO= 000030
C$GPRI= 000040
C$INIT= 000011
C$INLP= 000020
C$MANI= 000050
C$MEM = 000031
C$MSG = 000023
C$OPEN= 000034
C$PNTB= 000014
C$PNTF= 000017
C$PNTS= 000016
C$PNTX= 000015
C$QIO = 000377
C$RDBU= 000007
C$REFG= 000047
C$RESE= 000033
C$REVI= 000003
C$RFLA= 000021
C$RPT = 000025
C$SEFG= 000046
C$SPRI= 000041
C$SVEC= 000037
C$TPRI= 000013
DATARD  003416 G
DATAWT  003414 G
DATCNT= 010000 G
DATRAT  003412 G
DEVTBL  002536 G
DFPTBL  002176 G
DFTSCH= 000040 G
DIA   = 100006 G
DIABLK= 003414 G
DIACNT= 000020 G
DIAGMC= 000000
DINT    002212 G
DINTM   031506
DLY   = 000020 G
DLY.C = 000020 G
DRI   = 100013 G
DROPDM  004712 G
DROPED  003541 G
DROPN   020210
DROPU   017770 G
DROPUA  020120
DRORTN  020176
DTAERM  005472 G
DTAER2  004773 G
DTAER3  005042 G
DTAER4  005104 G
DTAER5  005125 G
EF.CON= 000036 G
EF.NEW= 000035 G
EF.PWR= 000034 G
EF.RES= 000037 G
EF.STA= 000040 G
EINC    016316
END   = 177777 G
ENDERF= 003506
ENDFLG= 003546
ENDSP   031322
ENDSP1  031130
ENDSP2  031132
EOTFLG  003516 G
ERCVER  002207 G
ERLOG   003502 G
ERRREC  003505 G
ERS   = 100411 G
ERSFLG  003545 G
EVL   = 000004 G
EWMSG   013050
EWPRNT  002216 G
EWPT    031610
EWRTRY  013174 G
EXALL   007036 G
EXARTN  007330
EXCRTN  011466
EXCUTE  010774 G
EXPBOT  003532 G
EXSUB   007332 G
E$END = 002100
E$LOAD= 000035
FATSM   004467 G
FIRSTU  017666 G
FMT.C0= 000040 G
FMT.C1= 000100 G
FTLCNT  003320 G
FUNRM   004447 G
F$AU  = 000015
F$AUTO= 000020
F$BGN = 000040
F$CLEA= 000007
F$DU  = 000016
F$END = 000041
F$HARD= 000004
F$HW  = 000013
F$INIT= 000006
F$JMP = 000050
F$MOD = 000000
F$MSG = 000011
F$PROT= 000021
F$PWR = 000017
F$RPT = 000012
F$SEG = 000003
F$SOFT= 000005
F$SRV = 000010
F$SUB = 000002
F$SW  = 000014
F$TEST= 000001
GCMDA   007746 G
GENPAT  010356 G
GES   = 100017 G
GETSTM  005253 G
GIT     010734
GOWAIT  011470 G
GSCPK   002324 G
G$CNTO= 000200
G$DELM= 000372
G$DISP= 000003
G$EXCP= 000400
G$HILI= 000002
G$LOLI= 000001
G$NO  = 000000
G$OFFS= 000400
G$OFSI= 000376
G$PRMA= 000001
G$PRMD= 000002
G$PRML= 000000
G$RADA= 000140
G$RADB= 000000
G$RADD= 000040
G$RADL= 000120
G$RADO= 000020
G$XFER= 000004
G$YES = 000010
```

```
HAE     002206 G    L$APT   002036 G    L10005  006612      NINUSE= 177774 G    RAMR5H  003344
HAEM    031372      L$AU    024424 G    L10006  006634      NOINTM  004515 G    RAMSIZ  003406 G
HALTM   004136 G    L$AUT   002070 G    L10007  006656      NRDYM   024244      RAMWRT  002214 G
HELP  = 000000      L$AUTO  023660 G    L10010  010756      NSSRM   004365 G    RANB    003440 G
HERE    003340      L$CCP   002106 G    L10011  022242      NUMBM   031732      RANBC = 153624 G
HOE   = 100000 G    L$CLEA  024302 G    L10013  023656      NURTY1  005167 G    RANCMD  026366
HRDCNT  003310 G    L$CO    002032 G    L10014  024146      OFLINM  005223 G    RANDOM  003533 G
IBE   = 010000 G    L$DEPO  002011 G    L10015  024356      ONEFIL= 000001       RANP  = 000007 G
IDU   = 000040 G    L$DESC  002140 G    L10016  024422      OPFLAG  003552 G    RANRD   026426
IER   = 020000 G    L$DESP  002076 G    L10017  024516      OPP.C = 020000 G    RANS    003442 G
IE.C  = 000200 G    L$DEVP  002060 G    L10020  026104      O$APTS= 000000       RANSC = 032561 G
INIT10  022252      L$DISP  002124 G    L10021  024652      O$AU  = 000001       RANW    026526
INIT15  022604      L$DLY   002116 G    L10022  024676      O$BGNR= 000001       RANWR   026502
INIT16  022624      L$DTP   002040 G    L10023  024716      O$BGNS= 000001       RANWV   026514
INTFLG  003506 G    L$DTYP  002034 G    L10024  024736      O$DU  = 000001       RAN1    010736 G
INTPRI= 000340 G    L$DU    024360 G    L10025  024756      O$ERRT= 000000       RAN2    010740 G
IRE     003540 G    L$DUT   002072 G    L10026  024776      O$GNSW= 000001       RAN3    010742 G
IREC    002210 G    L$DVTY  002166 G    L10027  025016      O$POIN= 000001       RCVERM  031416
IRECM   031440      L$EF    002052 G    L10030  025036      O$SETU= 000001       RDF   = 104001 G
IREM    031531      L$ENVI  002044 G    L10031  025056      PASCNT  003260 G    RDR   = 104401 G
ISR   = 000100 G    L$ETP   002102 G    L10032  025076      PATCH   032030 G    RECCNT  003330 G
IXE   = 004000 G    L$EXP1  002046 G    L10033  025134      PATERN  003454 G    RECLOG  003501 G
I$AU  = 000041      L$EXP4  002064 G    L10034  025320      PATRO   010442 G    RECRED  006560
I$AUTO= 000041      L$EXP5  002066 G    L10035  026560      PATR1   010500 G    RECTAP  006674 G
I$CLN = 000041      L$HARD  030474 G    L10036  027246      PATR2   010520 G    RECUD   012310 G
I$DU  = 000041      L$HIME  002120 G    L10037  027412      PATR3   010530 G    REPEAT= 050256
I$HRD = 000041      L$HPCP  002016 G    L10040  027524      PATR4   010554 G    RERM    004644 G
I$INIT= 000041      L$HPTP  002022 G    L10041  030470      PATR5   010566 G    RETRY = 050254
I$MOD = 000041      L$HW    002176 G    L10042  030544      PATR6   010600 G    RETRYC  003474 G
I$MSG = 000041      L$ICP   002104 G    L10043  032030      PATR7   010620 G    REWRT   015710
I$PROT= 000040      L$INIT  022252 G    L10044  032240      PATR8   010744 G    RFBC    002660 G
I$PTAB= 000041      L$LADP  002026 G    L10046  032244      PATTBL  010420       RFCERM  004350 G
I$PWR = 000041      L$LAST  032234 G    MBR   = 100012 G    PATTM   031752       RFREC   002760 G
I$RPT = 000041      L$LOAD  002100 G    MEMOM   023554      PCMDWD  003432 G    RFUNR   002770 G
I$SEG = 000041      L$LUN   002074 G    MISCFG  003551 G    PIRE    002213 G    RLEXM   004404 G
I$SETU= 000041      L$MREV  002050 G    MOD.C0= 000400 G    PNT   = 001000 G    RNF   = 125401 G
I$SFT = 000041      L$NAME  002000 G    MOD.C1= 001000 G    PRI   = 002000 G    RNOPSC= 177740 G
I$SRV = 000041      L$PRIO  002042 G    MOD.C2= 002000 G    PRI00 = 000000 G    RNR   = 105401 G
I$SUB = 000041      L$PROT  022244 G    MOD.C3= 004000 G    PRI01 = 000040 G    RNYM    004600 G
I$TST = 000041      L$PRT   002112 G    MOVMSG  012224 G    PRI02 = 000100 G    RPF   = 105001 G
JLOC    003452 G    L$REPP  002062 G    MSGCNT= 000016 G    PRI03 = 000140 G    RPR   = 125001 G
JLOOP   003450 G    L$REV   002010 G    MSGPKA  002506 G    PRI04 = 000200 G    RPTCNT  003476 G
JMP   = 000040 G    L$RPT   020440 G    MSGPKT  002340 G    PRI05 = 000240 G    RPTFLG  003535 G
JMPMSG  031562      L$SOFT  030546 G    MSGPKO  002356 G    PRI06 = 000300 G    RPT1A   021302
JMPMS2  031762      L$SPC   002056 G    MSGPK1  002374 G    PRI07 = 000340 G    RPT1B   021357
JMP.C = 000040 G    L$SPCP  002020 G    MSGPK2  002412 G    PRXST   020212 G    RPT1C   021430
J$JMP = 000167      L$SPTP  002024 G    MSGPK3  002430 G    PTCMDS  030342       RPT1D   021501
LBRVEC  003466 G    L$STA   002030 G    MS.RFC= 000004 G    PWRFLG  003547 G    RPT1E   021727
LCSR    003464 G    L$SW    002204 G    MS.XS0= 000006 G    RAMDAT  003346 G    RPT1F   021605
LENMSK  003436 G    L$TEST  002114 G    MS.XS1= 000010 G    RAMDUM  014436 G    RPT1G   021656
LHERTZ  003472 G    L$TIML  002014 G    MS.XS2= 000012 G    RAMER   016324 G    RPT1I   022046
LOE   = 040000 G    L$UNIT  002012 G    MS.XS3= 000014 G    RAMFHR  005320       RPT1J   021757
LOG     016024 G    L10000  002202      NCMD.C= 177740 G    RAMHLD  003342       RPT1K   022037
LOOPCT  003537 G    L10001  002312      NCNT    003420 G    RAMIOP  005377       RRANV   002205 G
LOT   = 000010 G    L10002  005636      NCNT1   003422 G    RAMLIN  005462       RRBC    002620 G
LVECT   003470 G    L10003  006562      NEXTSP  030614      RAMM    031564       RRECL = 000020 G
L$ACP   002110 G    L10004  006570      NEXTU   017734 G    RAMPD   005450       RRREC   002740 G
```

SYMBOL TABLE

```
RRUNR   002750 G    TIME1   003444 G    T$TAGL= 177777      WRITEM  020364 G    $F$TRU= 000404
RRVM    031343       TIME2   003446 G    T$TAGN= 010047      WRR   = 105005 G    $F$UNT= 000130
RS1   = 001233 G     TOERM   004303 G    T$TEMP= 000000      WRREC   002720 G    $F$WHI= 000120
RS2   = 007622 G     TOOMM   004554 G    T$TEST= 000006      WRT   = 104005 G    $F$YES= 000402
RS3   = 000000 G     TRAPD4  003550 G    T$TSTM= 177777      WRTY    014736 G    $IFLEV= 177777
RTLE    014600 G     TRAP4   024274 G    T$TSTS= 000001      WRTYCT  003250 G    $ISK0 = 000001
RTLRTN  014734       TRKMSG  013133       T$$AU = 010017      WRTYER  003500 G    $ISK1 = 000001
RWCPK   002334 G     TSAM    004532 G    T$$AUT= 010014      WRTYFG  003477 G    $ISK2 = 000001
RWD   = 102010 G     TSBA  = 002456 G    T$$CLE= 010015      WRUNR   002730 G    $ISK3 = 000001
RWERR   003503 G     TSC.FC= 177717 G    T$$DAT= 010046      WSSR    012124 G    $ISK4 = 000001
R5SAVE  003460 G     TSC.TC= 177761 G    T$$DU = 010016      WTM   = 100011 G    $ISK5 = 000001
SCCNT   003270 G     TSDB    002456 G    T$$HAR= 010042      WTR   = 101011 G    $ISK6 = 000001
SCERM   004324 G     TSSR    002466 G    T$$HW = 010000      WTV   = 104105 G    $LOCTA= 177777
SCH   = 140004 G     TSSREG  003462 G    T$$INI= 010013      WTVERM  004260 G    $LSTIN= 000001
SCHBK   002446 G     TSVCT   002476 G    T$$MSG= 010003      WTYBRF  015320       $LSTTA= 000001
SCHCNT= 000010 G     TS.A16= 000400 G    T$$PC = 000001      WTYCMD  015314       $LVTAG= 050436
SEQEND  003644 G     TS.A17= 001000 G    T$$PRO= 010012      WTYWRD  015316       $NESTL= 177777
SETCH   006772 G     TS.NBA= 002000 G    T$$PTA= 010045      X$ALWA= 000000       $NSK0 = 000120
SETRW   007016 G     TS.NXM= 004000 G    T$$RPT= 010011      X$FALS= 000040       $NSK1 = 000120
SETUP   010000 G     TS.OFL= 000100 G    T$$SOF= 010043      X$OFFS= 000400       $NSK2 = 000110
SFF   = 105010 G     TS.RMR= 010000 G    T$$SRV= 010010      X$TRUE= 000020       $NSK3 = 000110
SFPTBL  002204 G     TS.SC = 100000 G    T$$SUB= 010034      XO.BOT= 000002 G     $NSK4 = 000110
SFR   = 105410 G     TS.SPE= 020000 G    T$$SW = 010001      XO.EOT= 000001 G     $NSK5 = 000110
SRF   = 104010 G     TS.SSR= 000200 G    T$$TES= 010041      XO.LET= 020000 G     $NSK6 = 000110
SRR   = 104410 G     TS.UPE= 040000 G    T1      024520 G    XO.ONL= 000100 G     $SAVE = 000001
STAERM  005640 G     TS4ADR  030520       T1SWB   003542 G    XO.RLL= 010000 G     $SAVLE= 177777
STAER1  006152       TS4CL   002526 G    T1.1    024530      XO.RLS= 040000 G     $SELLE= 177777
STAER2  006330       TS4INT  002516 G    T1.10   025060      XO.TMK= 100000 G     $SSK0 = 050514
STAER3  006407       TS4IN0  006564 G    T1.11   025100      X1.EWN= 000010 G     $TAGLE= 177777
STAER4  006445       TS4IN1  006606 G    T1.12   025164      X2.OPM= 100000 G     $TAGNU= 050532
STAER5  006465       TS4IN2  006630 G    T1.2    024654      X3.DCK= 000010 G     $TEMP = 000402
STAER6  006274       TS4IN3  006652 G    T1.3    024700      X3.RNY= 157400 G     $TSK0 = 050513
STAER7  006244       TS4VCT  030535       T1.4    024720      ZROPAT  010504       $TSK1 = 050514
STAFLG  003546 G     T$ARGC= 000001       T1.5    024740      $BGNLE= 000002       $TSK2 = 050515
STREAM  003544 G     T$CODE= 023004       T1.6    024760      $BRJMP= 177777       $TSK3 = 050516
SVCGBL= 000000       T$ERRN= 000001       T1.7    025000      $BSK0 = 050254       $TSK4 = 050531
SVCINS= 000001       T$EXCP= 000000       T1.8    025020      $BSK1 = 050256       $TSK5 = 050530
SVCSUB= 000000       T$FLAG= 000041       T1.9    025040      $BSK2 = 050436       $TSK6 = 050524
SVCTAG= 000000       T$FREE= 032244       T2      026106 G    $ERFLG= 000400       $TSK7 = 050526
SVCTST= 000000       T$GMAN= 000000       T3      026562 G    $F$AND= 000310       $U    = 000403
SWBFLG  003536 G     T$HILI= 000010       T4      027250 G    $F$BAD= 000401       $$ARGC= 000000
SWB.C = 010000 G     T$LAST= 000001       T5      027414 G    $F$BLA= 000170       $$BYTE= 000403
SYMD  = 000007       T$LOLI= 000000       T5WEOT  030364       $F$CAS= 000150       $$CASE= 000000
SYMS  = 000007       T$LSYM= 010000       T6      027526 G    $F$DEC= 000220       $$DST = 000000
S$LSYM= 010000       T$LTNO= 000006       UAM   = 000200 G    $F$DO = 000340       $$ELOC= 000402
TAPCAP  022157       T$NEST= 177777       UNL   = 100412 G    $F$FAL= 000405       $$ERFL= 000000
TCCRA   013154       T$NS0 = 000000       UNREC   003504 G    $F$GOO= 000400       $$FLAG= 000001
TCC0    013306 G     T$NS1 = 000005       URERM   004666 G    $F$IF = 000110       $$FROM= 000000
TCC1    013330 G     T$NS2 = 000002       VFEXC   016642 G    $F$INC= 000210       $$INH = 000403
TCC2    013346 G     T$PCNT= 000000       VFISU   017126 G    $F$LOO= 000200       $$LOC = 027701
TCC3    013514 G     T$PTAB= 010045       VFYCNT  003300 G    $F$NAM= 000160       $$LOCN= 000000
TCC4    013536 G     T$PTHV= 000001       VFYDAT  016530 G    $F$NO = 000403       $$REG = 177777
TCC5    014256 G     T$PTNU= 000001       VFYFLG  003534 G    $F$OR = 000320       $$RETU= 000000
TCC6    014360 G     T$SAVL= 177777       VFY.C = 000100 G    $F$RTI= 000350       $$RTN1= 000000
TCC7    014414 G     T$SEGL= 177777       WLKZRO  010534       $F$RTN= 000300       $$RTN2= 000000
TC2RTN  013512       T$SIZE= 000004       WRBC    002560 G    $F$SEL= 000140       $$SRC = 000000
TC4STO  003530 G     T$SUBN= 000000       WRECL = 000020 G    $F$THE= 000330       $$TGSV= 000000
```

```
PARAMETER CODING        MACRO M1200  28-MAR-85 15:28  PAGE 114-4
SYMBOL TABLE                                                    SEQ 0182

$$TGS1= 000000          $$TGS2= 000000          $$TO  = 000000          $$$TAG= 050000

. ABS.  032244      000
        000000      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  58622 WORDS  ( 229 PAGES)
DYNAMIC MEMORY:  19748 WORDS  ( 75 PAGES)
ELAPSED TIME:  00:46:34
CZTKIB.BIN,CZTKIB/-SP/CR=SVC/ML,SPMACJ/ML,CZTKIB
```