```
                        .REM &

                                IDENTIFICATION
                                ..... .......

                        PRODUCT CODE:   AC-T783A-MC

                        PRODUCT NAME:   CZTKIAO TK25 DATA RELIABILITY TEST

                        PRODUCT DATE:   15 - MARCH - 1984

                        MAINTAINER:     MAGTAPE DIAGNOSTIC ENGINEERING

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

USER'S GUIDE

```
                                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)
```

```
152
153
154
155                             GLOSSARY
156                             --------
157
158                             ACT                          AUTOMATED COMPUTER TEST SYSTEM
159
160                             APT                          AUTOMATED PRODUCT TEST SYSTEM
161
162                             BYTE/RECORD/FILE COUNT       IS STORED IN THE 4TH  WORD OF THE COMMAND PACKET
163                             BRF                          AND IT'S  USE BY THE TK25 DEPENDS ON THE TYPE OF
164                                                          COMMAND.
165
166                             CMD                          TK25 COMMAND
167
168                             COMMAND PACKET               FOUR  WORD   PACKET  IN  THE  CPU  MEMORY  WHICH
169                             CMDPKT                       CONTAINS ALL INFORMATION  NEEDED BY THE  TK25 TO
170                                                          EXECUTE A COMMAND.
171
172                             EXTENDED STATUS              FOUR WORDS OF TK25 STATUS WHICH  ARE TRANSFERRED
173                                                          AS PART OF THE MESSAGE PACKET  AT THE COMPLETION
174                                                          OF A COMMAND.
175
176                             MESSAGE PACKET               SEVEN WORD PACKET IN THE  CPU MEMORY  INTO WHICH
177                                                          THE TK25  STORES  STATUS AT  THE COMPLETION OF A
178                                                          COMMAND.
179
180                             PC                           PROGRAM COUNTER
181
182                             PSW                          PROCESSOR STATUS WORD
183
184                             RESIDUAL FRAME COUNT         THIS COUNT IS  PART OF  THE  MESSAGE  PACKET AND
185                             RFC                          CONTAINS  THE   NUMBER  OF   BYTES/RECORDS/FILES
186                                                          REMAINING TO BE PROCESSED AT THE COMPLETION OF A
187                                                          COMMAND.
188
189                             SPECIAL CONDITION            TSS4 BIT15.  WHEN SET.  INDICATES THAT  THE LAST
190                             SPEC COND                    COMMAND DID NOT COMPLETE WITHOUT INCIDENT.
191
192                             TERMINATION CLASS CODE       THREE BIT CODE IN  THE TSSR WHICH  INDICATES THE
193                             TCC                          THE TYPE OF COMMAND TERMINATION.
194
195                             TSBA                         TAPE SYSTEM BUS ADDRESS REGISTER.
196
197                             TSDB                         TAPE SYSTEM DATA BUFFER REGISTER.
198
199                             TSSR                         TAPE SYSTEM STATUS REGISTER.
200
201                             XST0                         EXTENDED STATUS REGISTER 0
202
203                             XST1                         EXTENDED STATUS REGISTER 1
204
205                             XST2                         EXTENDED STATUS REGISTER 2
206
```

```
208
209
210
211                             XST3                    EXTENDED STATUS REGISTER 3
212
213                             XXDP+                   XXDP+  IS A  "CATCH-ALL"  NAME FOR  A  GROUP  OF
214                                                     PDP-11   DIAGNOSTIC    PACKAGES   AVAILABLE   ON
215                                                     MULTIMEDIA.
216
```

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

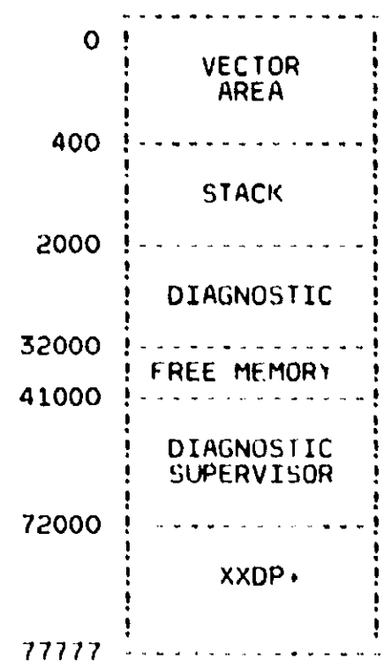
1.1.1 FUNCTIONAL DESCRIPTION -

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 -



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.

```
424
425
426
427
428                                     EXAMPLE OF A TAPE PASS PRINT:
429
430                                     CZTKIA SFT ERR  00009 ON UNIT 00 TST 002 SUB 000 PC: 012100
431                                     RECOVERABLE ERROR
432                                     WRT CMD FAILED - UNIT 0   PASS:    1   RECORD:    6
433                                     PREVIOUS CMD WAS  WRT
434                                     CMDPKT  TSBA    RFC     TSSR   TCC
435                                     100205  002406  000000  100210  4
436                                     026600
437                                     000000
438                                     003107
439                                     XST0    XST1    XST2    XST3
440                                     000350  000002  100400  000000
441                                     SUSPECT BAD SPOT AFTER 1 RETRY, 2 REPEAT
442                                     SUSPECT BAD SPOT AFTER 2 RETRY, 1 REPEAT
443                                     SUSPECT BAD SPOT AFTER 3 RETRY, 1 REPEAT
444                                     SUSPECT BAD SPOT AFTER 4 RETRY, 3 REPEAT
445                                     RETRY FAILED ON BAD SPOT...ERASED!
446                                     SUSPECT BAD SPOT AFTER 1 RETRY, 1 REPEAT
447
448                                     CZTKIA SFT ERR  00009 ON UNIT 00 TST 002 SUB 000 PC: 012100
449                                     RECOVERABLE ERROR
450                                     WRT CMD FAILED - UNIT 0   PASS:    1   RECORD:10210
451                                     PREVIOUS CMD WAS  WRT
452                                     CMDPKT  TSBA    RFC     TSSR   TCC
453                                     100205  002406  000000  100210  4
454                                     026600
455                                     000000
456                                     004000
457                                     XST0    XST1    XST2    XST3
458                                     000350  000002  100010  000000
459                                     RECOVERED ON RETRY # 1
460                                     ↑C
461                                     DR>PRI
462                                     UNIT 0   PASS:    1   RECORD:10210
463                                     BYTES WRITTEN    0,272,279,691
464                                     BYTES READ REV   0,301,123,654
465                                     BYTES READ REV   0,301,120,381
466                                                        WRT    RDR    RDF
467                                     RECOVERABLE ERRORS      1      0      0
468                                     UNRECOVERABLE ERRORS    0      0      0
469                                     WRITE RETRIES           3
470
471                                      2 BAD SPOTS THIS TAPE PASS PRECEDING RECORD #:
472                                               6      6
473                                     SPEC COND   HARD   FATAL   COMPARE
474                                           2      0      0       0
475                                     DR>
```

THIS EXAMPLE SHOWS THAT RECORD 5 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 -

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

1.2.2 SOFTWARE REQUIREMENTS -

- DIAGNOSTIC SUPERVISOR

1.3 RELATED DOCUMENTS AND STANDARDS

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

744
745
746
747 2.2.8 CONTROLLER RAM DUMP -
748
749 THIS QUESTION ALLOWS YOU TO CHOOSE WHETHER A DUMP OF THE
750 CONTOLLER'S RAM WILL OCCUR WITH EACH ERROR.
751
752
753
754 2.2.9 ENABLE EARLY WARNING MESSAGES -
755
756 THIS QUESTION ALLOWS YOU TO ENABLE OR DISABLE THE PRINTING OF A
757 MESSAGE EACH TIME A WRITE ERROR OCCURS AND THE EARLY WARNING (EW)
758 BIT IS SET IN EXTENDED STATUS REGISTER 1.
759
760
761
762 2.2.10 CHANGE CMD SEQUENCE -
763
764 THIS QUESTION WILL ALLOW YOU TO CHANGE THE COMMAND SEQUENCE USED
765 IN TEST 6.
766
767 NOTE: THIS QUESTION SHOULD BE ANSWERED (N) UNLESS AN OPERATOR
768 SELECTED SEQUENCE IS TO BE EXECUTED. IF THIS QUESTION IS
769 ANSWERED (N), NO MORE QUESTIONS WILL BE ASKED. IF THIS QUESTION
770 IS ANSWERED (Y), THE FOLLOWING QUESTIONS MUST BE ANSWERED OR
771 DEFAULTED WITH A CR>:

```
773
774
775
776
777
778     CHARACTERISTICS CODE (O) 40 ?   (0,20,40,200) (OCTAL)
779     CMD/2 (D) 13 ?                    (1-27)    (DECIMAL)
780     BRF COUNT (D) 1 ?                 (1-4K)    (DECIMAL)
781     # OF OPERATIONS (D) 1 ?          (1-32K)   (DECIMAL)
782     PATTERN (D) 7 ?                   (0-8)     (DECIMAL)
783     CMD/3 (D) 4 ?                     (1-27)    (DECIMAL)
784     BRF COUNT (D) 4096 ?              (1-4K)    (DECIMAL)
785     # OF OPERATIONS (D) 11719 ?      (1-32K)   (DECIMAL)
786     PATTERN (D) 7 ?                   (0-8)     (DECIMAL)
787     CMD/4 (D) 7 ?                     (1-27)    (DECIMAL)
788     BRF COUNT (D) 4096 ?              (1-4K)    (DECIMAL)
789     # OF OPERATIONS (D) 11719 ?      (1-32K)   (DECIMAL)
790     PATTERN (D) 7 ?                   (0-8)     (DECIMAL)
791     CMD/5 (D) 2 ?                     (1-27)    (DECIMAL)
792     BRF COUNT (D) 4096 ?              (1-4K)    (DECIMAL)
793     # OF OPERATIONS (D) 11719 ?      (1-32K)   (DECIMAL)
794     PATTERN (D) 7 ?                   (0-8)     (DECIMAL)
795     CMD/6 (D) 13. ?                   (1-27)    (DECIMAL)
796     BRF COUNT (D) 1 ?                 (1-4K)    (DECIMAL)
797     # OF OPERATIONS (D) 1 ?          (1-32K)   (DECIMAL)
798     PATTERN (D) 7 ?                   (0-8)     (DECIMAL)
799     CMD/7 (D) 27. ?                   (1-27)    (DECIMAL)
800     BRF COUNT (D) 4096 ?              (1-4K)    (DECIMAL)
801     # OF OPERATIONS (D) 11719 ?      (1-32K)   (DECIMAL)
802     PATTERN (D) 7 ?                   (0-8)     (DECIMAL)
803     CMD/8 (D) 27. ?                   (1-27)    (DECIMAL)
804     BRF COUNT (D) 4096 ?              (1-4K)    (DECIMAL)
805     # OF OPERATIONS (D) 11719 ?      (1-32K)   (DECIMAL)
806     PATTERN (D) 7 ?                   (0-8)     (DECIMAL)
807
808     NOTE:  THE PROGRAM AUTOMATICALLY INSERTS A  CHARACTERISTICS  CODE
809     40  (SEE BELOW) AS THE FIRST COMMAND IN THE SEQUENCE TABLE.  IF A
810     DIFFERENT CHARACTERISTIC IS DESIRED, THE  OPERATOR  SHOULD  ENTER
811     THAT CHARACTERISTIC CODE.
812
813     A TOTAL OF 7 COMMANDS MAY BE  ENTERED  IN  ADDITION  TO  THE  SET
814     CHARACTERISTICS COMMAND.  IF THE OPERATOR WISHES TO USE LESS THAN
815     7 COMMANDS, AN END COMMAND  (27)  MUST  BE  ENTERED  AND  THEN  A
816     CONTROL Z (Z) CAN BE ENTERED TO TERMINATE SOFTWARE DIALOGUE.
```

2.2.11 COMMAND LIST FOR USE IN SOFTWARE DIALOGUE. -

CODE	COMMAND	DESCRIPTION
1 =	DRI	DRIVE INITIATE.
2 =	RDF	READ FORWARD.
3 =	RDR	READ REVERSE.
4 =	WRT	WRITE.
5 =	WTV	WRITE/VERIFY. (WRITE N RECORDS; SPACE REVERSE N RECORDS; READ FORWARD AND CHECK N RECORDS.)
6 =	SRF	SPACE RECORDS FORWARD.
7 =	SRR	SPACE RECORDS REVERSE.
8 =	RNR	READ NEXT REVERSE, IE. SPACE FWD, READ REV.
9 =	RNF	READ NEXT FORWARD, IE. READ FWD, SPACE REV.
10 =	RPF	READ PREVIOUS FWD, IE. SPACE REV, READ FWD.
11 =	RPR	READ PREVIOUS REV, IE. READ REV, SPACE FWD.
12 =	WRR	WRITE RETRY.
13 =	RWD	REWIND.
14 =	MBR	MESSAGE BUFFER RELEASE.
15 =	WTM	WRITE TAPE MARK.
16 =	WTR	WRITE TAPE MARK RETRY.
17 =	SFF	SPACE FILES FORWARD.
18 =	SFR	SPACE FILES REVERSE.
19 =	GES	GET EXTENDED STATUS.
20 =	ERS	ERASE 3 INCHES OF TAPE.
21 =	UNL	UNLOAD.
22 =	CLN	CLEAN TAPE
23 =	SCH	SET DEVICE CHARACTERISTIC. WHERE BRF=200, 40, 20, 0. 200 = ENABLE SKIP TAPE MARKS STOP (STOP AT LOGICAL EOT) 40 = ENABLE ATTENTION INTERRUPTS. 20 = ENABLE MESSAGE BUFFER RELEASE INTERRUPTS. SEE TK25 PROGRAMMING SPECIFICATION FOR DESCRIPTION.
24 =	NOT USED	
25 =	JMP	JUMP TO THE NTH COMMAND IN THE COMMAND SEQUENCE TABLE, WHERE N IS DEFINED IN THE BRF FIELD. THE NUMBER OF JUMPS IS ENTERED IN THE NUMBER OF OPERATIONS FIELD.
26 =	DLY	DELAY "N" MILISECONDS WHERE N IS DEFINED IN THE NUMBER OF OPERATIONS. THIS DELAY IS USED BETWEEN EACH EXECUTABLE COMMAND.
27 =	END	END OF COMMAND SEQUENCE.

862
863
864
865 2.2.12 DATA PATTERN LIST FOR USE IN SOFTWARE DIALOGUE -
866
867 PATTERN # DESCRIPTION.
868
869 0 INCREMENTING PATTERN. 0 - 377.
870 1 ALL 1'S PATTERN.
871 2 ALL 0'S PATTERN.
872 3 1 BIT WALKING FROM R TO L IN A FIELD OF 0'S.
873 4 0 BIT WALKING FROM R TO L IF A FIELD OF 1'S.
874 5 ALTERNATING 1 AND 0 BITS WITH ALTERNATE BYTES COMPLIMENTED. (125/25
875 6 ALTERNATING BYTES OF 000 AND 377.
876 7 RANDOM DATA PATTERN.
877 8 NO PATTERN GENERATION.
878
879
880
881 2.3 EXAMPLES OF SOFTWARE DIALOGUE
882
883 2.3.1 BASIC FUNCTION AND DATA RELIABILITY WITH ALL ERROR
884 REPORTING ENABLED -
885
886 A) RECEIVE PROMPT (DR>)
887 B) ENTER STA/TES:1-2<CR>
888 C) ANSWER HARDWARE QUESTIONS.
889 D) PROCEED WITH THE FOLLOWING DIALOGUE:
890
891 CHANGE SW (L) ? Y<CR>
892 CLEAR COUNTERS (L) N ? Y<CR>
893 RESET RANDOM VARIABLES (L) N ? N<CR>
894 HALT AFTER EACH CMD (L) N ? N<CR>
895 PRINT SOFT ERRORS (L) N ? Y<CR>
896 INHIBIT RECOVERY (L) N ? N<CR>
897 BAD TAPE SPOT DETECT (L) Y ? Y<CR>
898 DISABLE INTERRUPTS (L) N ? N<CR>
899 INHIBIT RFC ERROR REPORT (L) N ? N<CR>
900 CONTROLLER RAM DUMP (L) N ? N<CR>
901 ENABLE EARLY WARNING MESSAGES (L) N ? N<CR>
902 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.

1107
1108
1109
1110 3.1.5 SPECIAL CONDITION ERRORS: -
1111
1112 IF, DURING EXECUTION, AN INCIDENT OCCURS FORCING THE TSSR SPECIAL
1113 CONDITION BIT TO SET, THE PROGRAM WILL SELECT ONE OF 8 ERROR
1114 HANDLING ROUTINES, DEPENDING ON THE TERMINATION CLASS CODE.
1115
1116 THE TERMINATION CLASS CODES IN THE TSSR ARE PROCESSED AS FOLLOWS
1117 WHEN SPECIAL CONDITION IS SET:
1118
1119
1120
1121 3.1.5.1 ERROR 5 - TERMINATION CLASS CODE 0, UNDEFINED SPECIAL
1122 CONDITION -
1123
1124 THE ERROR IS REPORTED, A HARD ERROR IS LOGGED AND THE PROGRAM
1125 PROCEEDS NORMALLY.
1126
1127
1128
1129 3.1.5.2 ERROR 6 - TERMINATION CLASS CODE 1, ATTENTION CONDITION
1130 -
1131
1132 THIS TCC INDICATES THAT THE DRIVE HAS UNDERGONE A STATUS CHANGE
1133 SUCH AS GOING OFFLINE OR COMING ONLINE. THIS IS A FATAL DEVICE
1134 ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS
1135 THE IDU OPTION IS USED.
1136
1137
1138
1139 3.1.5.3 ERROR 7 - TERMINATION CLASS CODE 2, TAPE STATUS ALERT -
1140
1141 A STATUS CONDITION HAS BEEN ENCOUNTERED THAT MAY HAVE
1142 SIGNIFICANCE TO THE PROGRAM. BITS OF INTEREST INCLUDE TMK, RLS,
1143 LET, RLL, EOT. ACTION TAKEN DEPENDS ON THE TEST BEING EXECUTED.
1144 IF THE CONDITION IS UNEXPECTED, THE ERROR IS REPORTED AND A HARD
1145 ERROR IS LOGGED.
1146
1147
1148
1149 3.1.5.4 ERROR 8 - TERMINATION CLASS CODE 3, FUNCTION REJECT -
1150
1151 THE SPECIFIED FUNCTION WAS NOT INITIATED. BITS OF INTEREST
1152 INCLUDE RMR, OFL, VCK, BOT, ILC, WLE, ILA, AND NBA. THIS IS A
1153 FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST
1154 CYCLE UNLESS THE IDU OPTION IS USED.

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.

```
1254
1255
1256
1257                    4.0  PERFORMANCE REPORT
1258
1259
1260                        UNIT X   PASS:XXXXX  RECORD:XXXXX
1261                        BYTES WRITTEN  XXX,XXX,XXX,XXX
1262                        BYTES READ REV XXX,XXX,XXX,XXX
1263                        BYTES READ FWD XXX,XXX,XXX,XXX
1264                                          WRT       RDR      RDF
1265                        RECOVERABLE ERRORS    XXXXX     XXXXX    XXXXX
1266                        UNRECOVERABLE ERRORS  XXXXX     XXXXX    XXXXX
1267
1268                        SPEC COND   HARD   FATAL   COMPARE
1269                           XXXXX   XXXXX   XXXXX   XXXXX
1270
1271
1272
1273
1274                    5.0  TEST SUMMARIES
1275
1276                    5.1  TEST 1 - BASIC FUNCTIONS.
1277
1278                    EXECUTES AND VERIFIES CORRECT COMPLETION OF ALL TK25 FUNCTIONS.
1279
1280                        SUBTEST 1   - SET CHAR, DRIVE INIT, GET STATUS.
1281                                    . SET CHARACTERISTIC 200. (ENABLES SKIP TAPE MARKS STOP)
1282                                    . DRIVE INITIATE.
1283                                    . SET CHARACTERISTIC  20. (ENABLES MESSAGE BUFF RELEASE INTERRUPTS)
1284                                    . GET STATUS
1285                                    . SET CHARACTERISTIC  40. (ENABLES ATTENTION INTERRUPTS)
1286
1287                        SUBTEST 2   - REWIND.
1288                                    . REWIND.
1289                                    . REWIND AT BOT.
1290
1291                        SUBTEST 3   - WRITE/VERIFY.
1292                                    . WRITE/VERIFY PATTERN 1.
1293                                    . WRITE/VERIFY PATTERN 2.
1294                                    . WRITE/VERIFY PATTERN 3.
1295                                    . WRITE/VERIFY PATTERN 4.
1296                                    . WRITE/VERIFY PATTERN 5.
1297                                    . WRITE/VERIFY PATTERN 6.
1298                                    . WRITE/VERIFY PATTERN 0.
1299
1300                        SUBTEST 4     WRITE TAPE MARK, ERASE.
1301                                    . WRITE TAPE MARK.
1302                                    . WRITE 10 RECORDS
1303                                    . ERASE 10 TIMES
1304                                    . WRITE TAPE MARK.
1305                                    . WRITE TAPE MARK RETRY.
1306
1307                        SUBTEST 5     SPACE FILES.
1308                                    . SPACE 2 FILES REVERSE.
```

```
1310
1311
1312
1313                                    * SPACE 2 FILES FORWARD.
1314                                    * SPACE 2 FILES REVERSE.
1315                                    * SPACE 2 FILES FORWARD.
1316
1317                                SUBTEST 6  - SPACE RECORDS.
1318                                    * REWIND.
1319                                    * SPACE 7 RECORDS FORWARD.
1320                                    * SPACE 7 RECORDS REVERSE.
1321                                    * SPACE 7 RECORDS FORWARD.
1322                                    * SPACE 7 RECORDS REVERSE.
1323
1324                                SUBTEST 7  - WRITE RETRY.
1325                                    * REWIND.
1326                                    * WRITE DATA.
1327                                    * WRITE RETRY.
1328
1329                                SUBTEST 8  - READ REV RETRY.
1330                                    * READ REVERSE.
1331                                    * READ NEXT REVERSE.
1332                                    * READ NEXT FORWARD.
1333
1334                                SUBTEST 9  - READ FWD RETRY.
1335                                    * READ FORWARD.
1336                                    * READ PREVIOUS FORWARD.
1337                                    * READ PREVIOUS REVERSE.
1338
1339                                SUBTEST 10 - CLEAN.
1340                                    * CLEAN.
1341                                    * REWIND.
1342
1343                                SUBTEST 11  - WRITE/VERIFY SWAPPED DATA BYTES.
1344                                    * WRITE/VERIFY EVEN LENGTH (RECORD 1).
1345                                    * WRITE/VERIFY ODD LENGTH (RECORD 2).
1346                                    * SET DATA BYTE SWAP.
1347                                    * WRITE/VERIFY EVEN LENGTH (RECORD 3).
1348                                    * WRITE/VERIFY ODD LENGTH (RECORD 4).
1349                                    * CLEAR DATA BYTE SWAP.
1350
1351                                SUBTEST 12  - READ SWAPPED DATA BYTES.
1352                                    * READ REV RECORD 4.
1353                                    * READ REV RECORD 3.
1354                                    * SET DATA BYTE SWAP.
1355                                    * READ REV RECORD 2.
1356                                    * READ REV RECORD 1.
1357                                    * READ FWD RECORD 1.
1358                                    * READ FWD RECORD 2.
1359                                    * CLEAR DATA BYTE SWAP.
1360                                    * READ FWD RECORD 3.
1361                                    * READ FWD RECORD 4.
1362
```

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 -

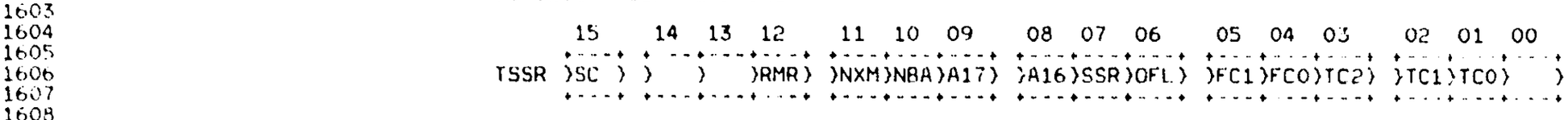
	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
TSBA	A15	A14	A13	A12	A11	A10	A09	A08	A07	A06	A05	A04	A03	A02	A01	A00
TSDB	P15	P14	P13	P12	P11	P10	P09	P08	P07	P06	P05	P04	P03	P02	P17	P16
TSSR	SC			RMR	NXM	NBA	A17	A16	SSR	OFL	FC1	FC0	TC2	TC1	TC0	
XST0	TMK	RLS	LET	RLL	WLE	NEF	ILC	ILA	MOT	ONL	IE	VCK		WLK	BOT	EOT
XST1	DLT			CRS	NER				TN3	TN2	TN1	TN0	EW		UNC	
XST2	OPM	DCF	DHF	SPD	0		1		ERROR ADDRESS LST SIGNIFICANT BYTE							
XST3	ERROR ADDRESS MOST SIGNIFICANT BYTE								OPI	REV	TCH	STP			RIB	

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):

   0 REWIND THE TAPE

   0 SPACE FORWARD ONE RECORD

   0 READ THE FIRST 256 BITS OF THE SECOND RECORD INTO CPU MEMORY STARTING AT ADDRESS 0.

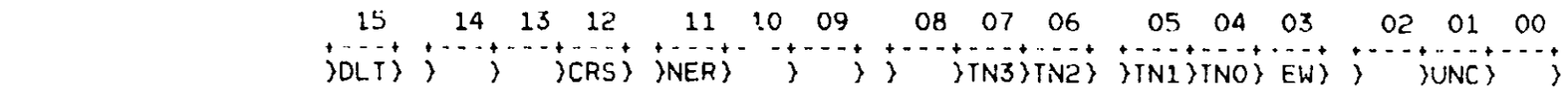
6.3.3 EXTENDED STATUS REGISTER 0 (XSTAT0) -

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

BIT	NAME	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

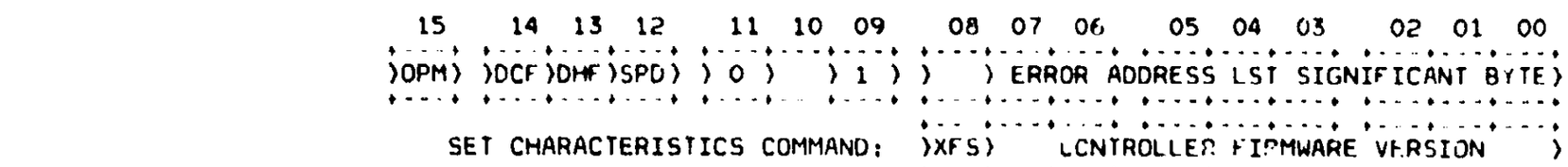
			. ANY MOTION COMMAND WITHOUT A CLEAR VOLUME CHECK WHILE THE TRANSPORT WAS OFF-LINE. . A WRITE COMMAND TO A WRITE PROTECTED TAPE.
09	ILC	3	ILLEGAL COMMAND: ANY COMMAND THAT CONTAINS CODES IN EITHER THE COMMAND FIELD OR MODE FIELD THAT ARE NOT SUPPORTED BY THE TAPE TRANSPORT.
08	ILA	3	ILLEGAL ADDRESS: 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 AT 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.

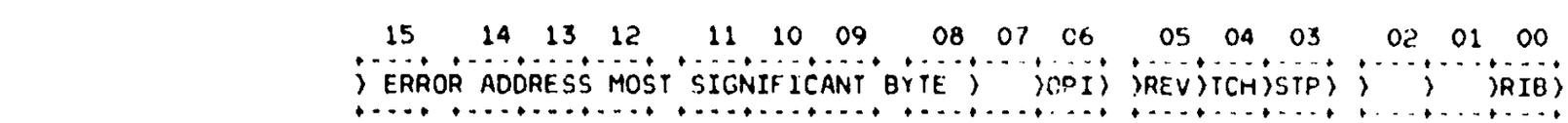
6.3.5 EXTENDED STATUS REGISTER 2 (XSTAT2) -



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) -



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	5	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	5/6	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.

&

```
                                .TITLE PROGRAM HEADER AND TABLES
                                .SBTTL  PROGRAM HEADER
                                
                                        .ENABL  ABS,AMA
                        002000  .       *       2000
002000                          BGNMOD
                                
                                ;++
                                ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
                                ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
                                ;--
                                
                                        POINTER BGNRPT,BGNSW,BGNSFT,BGNAU,BGNDU,BGNSETUP
                                
                                
                                        HEADER  CZTKI,A,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  101                                                             .ASCII  /A/
002011                          L$DEPO::                ;0
002011  060                                                             .ASCII  /0/
002012                          L$UNIT::                ;NUMBER OF UNITS
002012  000001                                                          .WORD   1$PTHV
002014                          L$TIML::                ;LONGEST TEST TIME
002014  005000                                                          .WORD   5000
002016                          L$HPCP::                ;POINTER TO H.W. QUES.
002016  027750                                                          .WORD   L$HARD
002020                          L$SPCP::                ;POINTER TO S.W. QUES.
002020  030022                                                          .WORD   L$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$LADP::                ;DIAG. END ADDRESS
002026  031510                                                          .WORD   L$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
```

```
002046                          L$EXP1::                ;EXPANSION WORD
002046  000000                                                          .WORD   0
002050                          L$MREV::                ;SVC REV AND EDIT #
002050     003                                                          .BYTE   C$REVISION
002051     003                                                          .BYTE   C$EDIT
002052                          L$EF::                  ;DIAG. EVENT FLAGS
002052  000000                                                          .WORD   0
002054  000000                                                          .WORD   0
002056                          L$SPC::
002056  000000                                                          .WORD   0
002060                          L$DEVP::                ; POINTER TO DEVICE TYPE LIST
002060  002166                                                          .WORD   L$DVTYP
002062                          L$REPP::                ;PTR. TO REPORT CODE
002062  020052                                                          .WORD   L$RPT
002064                          L$EXP4::
002064  000000                                                          .WORD   0
002066                          L$EXP5::
002066  000000                                                          .WORD   0
002070                          L$AUT::                 ;PTR. TO ADD UNIT CODE
002070  023710                                                          .WORD   L$AU
002072                          L$DUT::                 ;PTR. TO DROP UNIT CODE
002072  023644                                                          .WORD   L$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$ICP::                 ;PTR. TO INIT CODE
002104  021664                                                          .WORD   L$INIT
002106                          L$CCP::                 ;PTR. TO CLEAN-UP CODE
002106  023602                                                          .WORD   L$CLEAN
002110                          L$ACP::                 ;PTR. TO AUTO CODE
002110  023160                                                          .WORD   L$AUTO
002112                          L$PRT::                 ;PTR. TO PROTECT TABLE
002112  021656                                                          .WORD   L$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
2041
```

```
                        .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
002124                  L$DISPATCH::
002124  024004                                                          .WORD   T1
002126  025372                                                          .WORD   T2
002130  026046                                                          .WORD   T3
002132  026524                                                          .WORD   T4
002134  026670                                                          .WORD   T5
002136  027002                                                          .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.
                        .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
                        ENDSW
002312          L10001:
002312                  ENDMOD
```

```
2166
2167                            .TITLE GLOBAL AREAS
2168                            .SBTTL  GLOBAL EQUATES SECTION
2177
2178 002312                            BGNMOD
2179
2180                            ;++
2181                            ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
2182                            ; ARE USED IN MORE THAN ONE TEST.
2183                            ;--
2184
2185 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          PRIO5== 240
                000200          PRIO4== 200
                000140          PRIO3== 140
                000100          PRIO2== 100
                000040          PRIO1== 40
                000000          PRIO0== 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
2186
2194                            ; REGISTER USAGE.
2195                            ;
2196                            ;       R0 - PASSES PARAMETERS TO/FROM DIAGNOSTIC SUPERVISOR.
2197                            ;       R1 - COMMAND SEQUENCE TABLE POINTER.
2198                            ;       R2 - GENERAL PURPOSE REGISTER.
2199                            ;       R3 - GENERAL PURPOSE REGISTER.
2200                            ;       R4 - GENERAL PURPOSE REGISTER.
2201                            ;       R5 - CURRENT LOGICAL DEVICE NUMBER X 2.
2202                            ;       R6 - STACK POINTER.
2203                            ;       R7 - PROGRAM COUNTER.
2204
2205                            ;THE FOLLOWING ARE BIT DEFINITIONS FOR THE TSSR REGISTERS.
2206
2207            100000          TS.SC==100000           ;SPECIAL CONDITION BIT.
2208            040000          TS.UPE==40000           ;UNIBUS PARITY ERROR
2209            020000          TS.SPE==20000           ;SERIAL BUS PARITY ERROR.
2210            010000          TS.RMR==10000           ;REGISTER MODIFICATION REFUSED.
2211            004000          TS.NXM==4000            ;NON-EXISTENT MEMORY.
2212            002000          TS.NBA==2000            ;NEED BUFFER ADDRESS
2213            001000          TS.A17==1000            ;BUS ADDRESS BIT 17.
2214            000400          TS.A16==400             ;BUS ADDRESS BIT 16.
2215            000200          TS.SSP==200             ;UNIT READY BIT.
2216            000100          TS.OFL==100             ;OFF LINE.
2217            177717          TSC.FCC==177717         ;FATAL CLASS CODE MASK.
2218            177761          TSC.TCC==177761         ;TERMINATION CLASS CODE MASK.
2219
```

```
2221                            ;THE FOLLOWING ARE BIT DEFINITIONS FOR THE COMMAND WORD
2222
2223    100000                  ACK.C==100000           ;ACKNOWLEDGE BIT
2224    040000                  CVC.C==40000            ;CLEAR VOLUME CHECK.
2225    020000                  OPP.C==20000            ;OPPOSITE BIT
2226    010000                  SWB.C==10000            ;SWAP BYTE BIT
2227    004000                  MOD.C3==4000            ;MODE BIT 3
2228    004000                  BRF.C==4000             ;BYTE/RECORD/FILE COUNT FLAG BIT.  NOT USED
2229                                                    ;BY TK25 BUT USED INTERNALLY BY THIS PROGRAM ONLY.
2230    002000                  MOD.C2==2000            ;MODE BIT 2
2231    001000                  MOD.C1==1000            ;MODE BIT 1
2232    000400                  MOD.C0==400             ;MODE BIT 0
2233    000200                  IE.C==200               ;INTERRUPT ENABLE
2234    000100                  FMT.C1==100             ;FORMAT BIT 1
2235    000100                  VFY.C==100              ;WRITE VERIFY FLAG BIT.  INTERNAL USE ONLY.
2236                                                    ;NOT USED BY TK25.
2237    000040                  FMT.C0==40              ;FORMAT BIT 0.
2238    000040                  JMP.C==40               ;JUMP BIT - CORRECT THIS PROGRAM TO JUMP TO
2239                                                    ;A CERTAIN LOCATION IN THE COMMAND SEQUENCE
2240                                                    ;TABLE.  INTERNAL USE ONLY.
2241    000020                  CMD.C4==20              ;COMMAND BIT 4
2242    000020                  DLY.C==20               ;INSERT DELAY.  INTERNAL USE ONLY.
2243    000010                  CMD.C3==10              ;COMMAND BIT 3
2244    000004                  CMD.C2==4               ;COMMAND BIT 2
2245    000002                  CMD.C1==2               ;COMMAND BIT 1
2246    000001                  CMD.C0==1               ;COMMAND BIT 0
2247
2248                            ;       BIT DEFINITIONS FOR DEVICE CHARACTERISTICS.
2249
2250    000200                  CH.ESS==200             ;ENABLE SKIP TAPE MARKS STOP (STOP AT LOGICAL EOT).
2251    000040                  CH.EAI==40              ;ENABLE ATTENTION INTERRUPTS.
2252    000020                  CH.ERI==20              ;ENABLE MESSAGE BUFFER RELEASE INTERRUPTS.
2253    000040                  DFTSCH==CH.EAI          ;DEFAULT CHARACTERISTICS CODE.
2254
2255                            ;THE FOLLOWING INDICATES THE RELATIVE POSITIONS OF THE STATUS WORDS
2256                            ;IN THE MESSAGE BUFFER.
2257
2258    000004                  MS.RFC==4               ;RESIDUAL FRAME COUNT.
2259    000006                  MS.XS0==6               ;EXT STATUS REG 0
2260    000010                  MS.XS1==10              ;EXT STATUS REG 1
2261    000012                  MS.XS2==12              ;EXT STATUS REG 2
2262    000014                  MS.XS3==14              ;EXT STATUS REG 3
2263
2264                            ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 0.
2265
2266    100000                  X0.TMK==100000          ;TAPE MARK.
2267    040000                  X0.RLS==40000           ;RECORD LENGTH SHORT.
2268    020000                  X0.LET==20000           ;LOGICAL EOT.
2269    010000                  X0.RLL==10000           ;RECORD LENGTH LONG.
2270    000100                  X0.ONL==100             ;ON LINE BIT.
2271    000002                  X0.BOT==2               ;BOT BIT.
2272    000001                  X0.EOT==1               ;EOT BIT.
2273
2274                            ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 1.
2275
2276    000010                  X1.EWN==BIT3
2277
```

```
                                ;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==PRIO7                   ;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
```

```
2335          104410                  SRR==   ACK.C!BRF.C!MOD.C0!CMD.C3
2336                                                  ;SPACE RECORD REVERSE
2337
2338          105401                  RNR==   ACK.C!BRF.C!MOD.C1!MOD.C0!CMD.C0
2339                                          ;READ REV RETRY1 - REREAD NEXT REVERSE, IE. SPACE FWD, READ REVERSE
2340
2341          125401                  RNF==   ACK.C!BRF.C!OPP.C!MOD.C1!MOD.C0!CMD.C0
2342                                          ;READ REV RETRY2 - REREAD NEXT FORWARD, IE.READ FORWARD, SPACE REVERSE
2343
2344          105001                  RPF==   ACK.C!BRF.C!MOD.C1!CMD.C0
2345                                          ;READ FWD RETRY1 - REREAD PREVIOUS FORWARD, IE. SPACE REVERSE, READ FORWARD
2346
2347          125001                  RPR==   ACK.C!BRF.C!OPP.C!MOD.C1!CMD.C0
2348                                          ;READ FWD RETRY2 - REREAD PREVIOUS REVERSE, IE. READ REVERSE, SPACE FORWARD
2349
2350          105005                  WRR==   ACK.C!MOD.C1!BRF.C!CMD.C2!CMD.C0
2351                                                  ;WRITE RETRY
2352
2353          102010                  RWD==   ACK.C!MOD.C2!CMD.C3
2354                                                  ;REWIND COMMAND
2355
2356          100012                  MBR==   ACK.C!CMD.C3!CMD.C1
2357                                                  ;MESSAGE BUFFER RELEASE
2358
2359          100011                  WTM==   ACK.C!CMD.C3!CMD.C0
2360                                                  ;WRITE TAPE MARK.
2361
2362          101011                  WTR==   ACK.C!MOD.C1!CMD.C3!CMD.C0
2363                                                  ;WRITE TAPE MARK RETRY.
2364
2365          105010                  SFF==   ACK.C!BRF.C!MOD.C1!CMD.C3
2366                                                  ;SPACE FILE FORWARD
2367
2368          105410                  SFR==   ACK.C!BRF.C!MOD.C0!MOD.C1!CMD.C3
2369                                                  ;SPACE FILE REVERSE
2370
2371          100017                  GES==   ACK.C!CMD.C0!CMD.C1!CMD.C2!CMD.C3
2372                                                  ;GET EXTENDED STATUS
2373
2374          100411                  ERS==   ACK.C!MOD.C0!CMD.C3!CMD.C0
2375                                                  ;ERASE 3 INCHES OF TAPE
2376
2377          100412                  UNL==   ACK.C!MOD.C0!CMD.C3!CMD.C1
2378                                                  ;UNLOAD COMMAND
2379
2380          101012                  CLN==   ACK.C!MOD.C1!CMD.C3!CMD.C1
2381                                                  ;ERASE TAPE.
2382
2383          140004                  SCH==   ACK.C!CVC.C!CMD.C2      ;SET DEVICE CHARACTERISTICS.
2384
2385          100006                  DIA==   ACK.C!CMD.C2!CMD.C1     ;DIAGNOSTICS.
2386
2387          000040                  JMP==   JMP.C                   ;JUMP TO "N"TH COMMAND
2388
2389          000020                  DLY==   DLY.C                   ;DELAY "N" MS.
2390
2391          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         .       =       .+36177774      ;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         .       =       .+36177774      ;MUST BE ON MOD 4 BOUNDRY.
002324  100017                  GSCPK:: .WORD   GES

                                ;       MESSAGE BUFFER RELEASE COMMAND PACKET.

                 002330         .       =       .+36177774      ;MUST BE ON MOD 4 BOUNDRY.
002330  100012                  BRCPK:: .WORD   MBR

                                ;       REWIND COMMAND PACKET (USED IN ERROR RECOVERY ONLY)

                 002334         .       =       .+36177774      ;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  006552                  TS4INT:: TS4IN0                 ;DEVICE 0.
002520  006560                           TS4IN1                 ;DEVICE 1.
002522  006566                           TS4IN2                 ;DEVICE 2.
002524  006574                           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
```

```
2495                                    ;       COUNTER AREA.
2496
2497            002560                  CNTBGN=.
2498 002560                             WRBC::  .BLKW   20              ;BYTES WRITTEN.
2499 002620                             RRBC::  .BLKW   20              ;BYTES READ REV.
2500 002660                             RFBC::  .BLKW   20              ;BYTES READ FWD.
2501 002720                             WRREC:: .BLKW   4               ;RECOVERABLE WRITE ERRORS.
2502 002730                             WRUNR:: .BLKW   4               ;UNRECOVERABLE WRITE ERRORS.
2503 002740                             RRREC:: .BLKW   4               ;RECOVERABLE READ REV ERRORS.
2504 002750                             RRUNR:: .BLKW   4               ;UNRECOVERABLE READ REV ERRORS.
2505 002760                             RFREC:: .BLKW   4               ;RECOVERABLE READ FWDERRORS.
2506 002770                             RFUNR:: .BLKW   4               ;UNRECOVERABLE READ FWD ERRORS.
2507 003000                             BT0::   .BLKW   21.             ;UNIT 0 BAT TAPE SPOTS LOG
2508 003052                             BT1::   .BLKW   21.             ;UNIT 1 BAT TAPE SPOTS LOG
2509 003124                             BT2::   .BLKW   21.             ;UNIT 2 BAT TAPE SPOTS LOG
2510 003176                             BT3::   .BLKW   21.             ;UNIT 3 BAT TAPE SPOTS LOG
2511 003250                             WRTYCT:: .BLKW  4               ;WRITE RETRY COUNTER
2512 003260                             PASCNT:: .BLKW  4               ;PASS COUNT.
2513 003270                             SCCNT:: .BLKW   4               ;SPECIAL CONDITION COUNT.
2514 003300                             VFYCNT:: .BLKW  4               ;COUNT OF TK25 DATA COMPARE ERRORS.
2515 003310                             HRDCNT:: .BLKW  4               ;COUNT OF HARD ERRORS.
2516 003320                             FTLCNT:: .BLKW  4               ;COUNT OF FATAL ERRORS.
2517            003330                  CNTEND=.                        ;END OF STATICTICAL COUNTERS.
2518 003330                             RECCNT:: .BLKW  4               ;NUMBER OF RECORDS FROM BOT: CLEARED  ON REWIND
2519                                                                    ;AND WHEN RESTARTING OR CONTINUING TEST 2.
2520            000550                  CNTLEN==CNTEND-CNTBGN           ;LENGTH OF STATISTICAL COUNTER AREA.
2521
2522 003340      000                    HERE:   .BYTE 0                 ;TEST 3 ASCII SEMAPHORE
2523                                    ;       THE FOLLOWING ARE THE DEFINITIONS OF VARIABLES
2524                                    ;       USED BY THE PROGRAM.
2525
2526                                    .EVEN
2527 003342  000000                    RAMHLD:   .WORD  0              ;RAM ADDR HOLDER 1ST ADDRESS
2528 003344  000000                    RAMR5H:   .WORD  0              ;HOLDS R5 FOR LATER
2529 003346                             RAMDATA::.BLKW  16.             ;DATA READ FROM RAM PACKET OR MESSAGE BUF AREA
2530 003406  000000                    RAMSIZ:: .WORD  0               ;RAM DATA SIZE FOR PRAMPKT ROUTINE
2531 003410  000000                    CMPDAT:: .WORD  0               ;COUNTS # OF READS (TEST 3) BEFORE ALLOWING A DATA COMPARE
2532 003412  000003                    DATRAT:: .WORD  3               ;CONTROLS THE DATA COMPARE RATIO
2533 003414  000000                    DATAWT:: .WORD  0               ;WRITE BUFFER ADDRESS.
2534 003416  000000                    DATARD:: .WORD  0               ;READ BUFFER ADDRESS.
2535 003420  000000                    NCNT::    .WORD  0              ;STORAGE FOR VALUE OF N.
2536 003422  000000                    NCNT1::   .WORD  0              ;TEMP STORAGE FOR VALUE OF N.
2537 003424  000000                    BRFCNT:: .WORD  0               ;STORAGE FOR BPCR VALUE.
2538 003426  177777                    CMDWRD:: .WORD  END             ;CONTAINS COMMAND WORD BEING EXECUTED PRESENTLY.
2539 003430  177777                    CMDSAV:: .WORD  END             ;SAVE LOCATION FOR CMD WORD DURING ERROR RECOVERY
2540 003432  177777                    PCMDWD:: .WORD  END             ;CONTAINS PREVOUS COMMAND WORD.
2541 003434  000000                    CMDLG::   .WORD  0              ;CURRENT COMMAND LOGGING CODE.
2542 003436  000000                    LENMSK:: .WORD  0               ;RANDOM WRITE LENGTH MASK, TO BE SET UP BY TESTS
2543 003440  153624                    RANB::    .WORD  153624         ;RANDOM # GENERATOR BASE.
2544 003442  032561                    RANS::    .WORD  32561          ;RANDOM # SAVE LOCATION.
2545 003444  000000                    TIME1::   .WORD  0              ;TIME COUNT 1.
2546 003446  000000                    TIME2::   .WORD  0              ;TIME COUNT 2.
2547 003450  000000                    JLOOP::   .WORD  0              ;JMP COMMAND LOOP COUNT.
2548 003452  000000                    JLOC::    .WORD  0              ;JMP COMMAND LOCATION COUNT.
2549 003454  000000                    PATERN:: .WORD  0               ;PATTERN SELECT CODE.
```

```
2551 003456  000000                CTCC::   .WORD  0              ;CURRENT TERMINATION CLASS CODE.
2552 003460  000000                R5SAVE:: .WORD  0              ;LOCATION FOR SAVING CURRENT DEVICE POINTER.
2553 003462  000000                TSSREG:: .WORD  0              ;CURRENT STATUS REGISTER.
2554         003414                DIABLK=.DATAWT                 ;WRITE BUFFER ALSO USED FOR DIAG CMD.
2555
2556                               ;       ERROR FLAG AREA, THESE FLAGS ARE CLEARED DURING INITIALIZATION AND
2557                               ;       AFTER EACH COMMAND IS COMPLETED.
2558
2559         003464                BGNFLG=.
2560 003464  000000                RETRYC:: .WORD  0              ;# OF RECOVERY ATTEMPTS EXECUTED.
2561 003466     000                RPTCNT:: .BYTE  0       ;WRITE REPEAT ON SAME SPOT CNTR: 4 PER WRITE RETRY
2562 003467     000                WRTYFG:: .BYTE  0              ;WRITE RETRY ON SAME SPOT IN PROGRESS FLAG
2563 003470     000                WRTYER:: .BYTE  0              ;WRITE RETRY ON SAME SPOT ERROR FLAG
2564 003471     000                RECLOG:: .BYTE  0              ;RECORD COUNT HAS BEEN UPDATED FOR THIS RECORD.
2565 003472     000                ERLOG::  .BYTE  0              ;DATA BYTES AND ERRORS HAVE BEEN LOGGED FOR THIS RECORD.
2566 003473     000                RWERR::  .BYTE  0              ;READ/WRITE ERROR HAS OCCURED.
2567 003474     000                UNREC::  .BYTE  0              ;UNRECOVERABLE ERROR HAS OCCURED.
2568 003475     000                ERRREC:: .BYTE  0              ;ERROR RECOVERY MODE.
2569                                        .EVEN
2570         003476                ENDERF=.
2571
2572                               ;       ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED DURING INITIALIZATION.
2573
2574 003476                        INTFLG:: .BLKW  4              ;INTERRUPT OCCURRED FLAGS FOR EACH DEVICE.
2575 003506                        EOTFLG:: .BLKW  4              ;EOT/BOT FLAGS FOR EACH DEVICE (XSTAT0).
2576 003516  000000                BTPT::   .WORD  0              ;BAD TAPE SPOT POINTER TO BT0-BT3 VIA BTADDR
2577 003520     000                EXPBOT:: .BYTE  0              ;BOT IS EXPECTED, DO NOT ABORT ON BOT/FUNC RTI.
2578 003521     000                RANDOM:: .BYTE  0              ;RANDOM EVERYTHING FLAG.
2579 003522     000                VFYFLG:: .BYTE  0              ;SET DURING WRITE/VERIFY COMMAND.
2580 003523     000                RPTFLG:: .BYTE  0              ;PERFORMANCE REPORT HAS BEEN REQUESTED.
2581 003524     000                SWBFLG:: .BYTE  0              ;ENABLES SWAP BYTE FUNCTION WHEN NOT EQUAL TO ZERO.
2582 003525     000                LOOPCT:: .BYTE  0              ;WRITE LOOP CONTROL (TEST 3)
2583 003526     000                IRE::    .BYTE  0              ;INHIBIT RESIDUAL FRAME COUNT ERROR REPORT.
2584 003527     000                DROPED:: .BYTE  0              ;CURRENT UNIT HAS BEEN DROPPED
2585 003530     000                T1SWB::  .BYTE  0              ;TEST1 SWAP BYTES FLAG
2586 003531     000                ALLEOT:: .BYTE  0              ;ALL UNITS @ EOT FLAG
2587 003532     000                STREAM:: .BYTE  0              ;INDICATES TEST ONE UNIT AT A TIME, COMPLETELY.
2588 003533     000                ERSFLG:: .BYTE  0              ;ERASE FLAG:  DO ERASE AFTER A SPACE REV TO DELETE
2589                                                              ;BADLY WRITTEN RECORD.  1 TO 4 ERASES LEAVING
2590                                                              ;A 3 TO 12 INCH GAP MAY RESULT.
2591                                       .EVEN
2592         003534                ENDFLG=.
2593
2594                               ;       ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED ONLY AFTER BEING CHECKED.
2595
2596 003534     000                STAFLG:: .BYTE  0              ;START FLAG - SET BY INIT CODE IF STARTING.
2597 003535     000                PWRFLG:: .BYTE  0              ;POWER FAILURE FLAG - SET ONLY DURING INIT.
2598 003536     000                TRAPD4:: .BYTE  0              ;TRAPED AT 4 FLAG
2599 003537     000                MISCFG:: .BYTE  0              ;MISCELLANEOUS FLAG
2600
```

```
2602                                    ;       OPERATOR FLAG SETTINGS PASSED BY DIAG. SUPERVISOR IN A 16 BIT WORD
2603                                    ;       SEE GLOBAL EQUATES SECTION FOR FLAG BIT LIST
2604
2605 003540  000000                     OPFLAG:: .WORD  0       ;READ ONLY OPERATOR FLAG WORD
2606                                            .EVEN
2607
2608                                            ;THE FOLLOWING IS THE COMMAND SEQUENCE TABLE.  THE TABLE
2609                                            ;HAS DEFAULT VALUES AT PROGRAM LOAD AS SHOWN.  THESE VALUES
2610                                            ;CAN BE UPDATED BY A TEST OR BY OPERATOR INPUT.
2611
2612 003542  140004                     CMDSEQ:: .WORD  SCH             ;SET CHARACTERISTICS.
2613 003544  000040                             .WORD   CH.EAI
2614 003546  000001                             .WORD   1
2615 003550  000000                             .WORD   0
2616 003552  102010                     CMDSE2:: .WORD  RWD             ;REWIND.
2617 003554  000001                             .WORD   1               ;BYTE COUNT.
2618 003556  000001                             .WORD   1               ;ONCE.
2619 003560  000007                             .WORD   RANP            ;PATTERN.
2620 003562  104005                             .WORD   WRT             ;WRITE.
2621 003564  010000                             .WORD   DATCNT          ;MAX BUFFER LENGTH.
2622 003566  035230                             .WORD   15000.          ;15000 RECORDS.
2623 003570  000007                             .WORD   RANP            ;PATTERN - RANDOM DATA
2624 003572  102010                             .WORD   RWD             ;REWIND
2625 003574  000001                             .WORD   1               ;BYTE COUNT
2626 003576  000001                             .WORD   1               ;ONE ITERATION
2627 003600  000007                             .WORD   RANP            ;RANDOM DATA
2628 003602  104001                             .WORD   RDF             ;READ FWD.
2629 003604  010000                             .WORD   DATCNT          ;MAX BUFFER LENGTH.
2630 003606  035230                             .WORD   15000.          ;15000 RECORDS.
2631 003610  000007                             .WORD   RANP            ;RANDOM DATA
2632 003612  102010                             .WORD   RWD             ;REWIND
2633 003614  000001                             .WORD   1               ;BYTE COUNT.
2634 003616  000001                             .WORD   1               ;ONCE.
2635 003620  000007                             .WORD   RANP            ;PATTERN.
2636 003622                                     .BLKW   4               ;EXTENSION TO HOLD 1 MORE CMD.
2637 003632  177777                     SEQEND:: .WORD  END             ;SOFT END OF SEQUENCE TABLE.
2638 003634  177777                             .WORD   END
2639 003636  177777                             .WORD   END
2640 003640  177777                             .WORD   END
2641 003642  177777                             .WORD   END             ;HARD END OF SEQUENCE TABLE.
```

```
2643                                    ;THE FOLLOWING IS THE TK25 COMMAND TABLE
2644
2645 003644  100013            CMDTBL:: .WORD  DRI             ;DRIVE INIT.
2646 003646  104001                     .WORD  RDF             ;READ FORWARD.
2647 003650  104401                     .WORD  RDR             ;READ REVERSE.
2648 003652  104005                     .WORD  WRT             ;WRITE
2649 003654  104105                     .WORD  WTV             ;WRITE/VERIFY. (WRITE ALL RECORDS, RDR AND
2650                                                            ;CHECK DATA ON ALL RECORDS, RDF AND
2651                                                            ;CHECK DATA ON ALL RECORDS.)
2652 003656  104010                     .WORD  SRF             ;SPACE "N" RECORDS FORWARD.
2653 003660  104410                     .WORD  SRR             ;SPACE "N" RECORDS REVERSE.
2654 003662  105401                     .WORD  RNR             ;READ NEXT REVERSE. I.E., SPACE FWD, READ REVERSE.
2655 003664  125401                     .WORD  RNF             ;READ NEXT FORWARD, I.E., READ FORWARD, SPACE REVERSE.
2656 003666  105001                     .WORD  RPF             ;READ PREVIOUS FORWARD. I.E., SPACE REVERSE, READ FORWARD
2657 003670  125001                     .WORD  RPR             ;READ PREVIOUS REVERSE. I.E., READ REVERSE, SPACE FORWARD
2658 003672  105005                     .WORD  WRR             ;WRITE RETRY.
2659 003674  102010                     .WORD  RWD             ;REWIND.
2660 003676  100012                     .WORD  MBR             ;MESSAGE BUFFER RELEASE
2661 003700  100011                     .WORD  WTM             ;WRITE TAPE MARK
2662 003702  101011                     .WORD  WTR             ;WRITE TAPE MARK RETRY.
2663 003704  105010                     .WORD  SFF             ;SPACE "N" FILES FORWARD.
2664 003706  105410                     .WORD  SFR             ;SPACE "N" FILES REVERSE.
2665 003710  100017                     .WORD  GES             ;GET EXTENDED STATUS.
2666 003712  100411                     .WORD  ERS             ;ERASE 3 INCHES OF TAPE.
2667 003714  100412                     .WORD  UNL             ;REWIND AND UNLOAD.
2668 003716  101012                     .WORD  CLN             ;CLEAR TAPE.
2669 003720  140004                     .WORD  SCH             ;SET CHARACTERISTICS.
2670 003722  100006                     .WORD  DIA             ;DIAGNOSTIC COMMAND.
2671 003724  000040                     .WORD  JMP             ;JUMP TO THE NTH COMMAND IN THE SEQUENCE.
2672 003726  000020                     .WORD  DLY             ;DELAY "N" MS.
2673 003730  177777                     .WORD  END             ;END OF COMMAND TABLE
2674
```

```
2676                                    ;       THE FOLLOWING TABLE CONTAINS THE ASCII FOR EACH COMMAND.
2677
2678 003732    104     122     111     CMDASC:: .ASCII /DRI/           ;DRIVE INIT.
2679 003735    122     104     106             .ASCII  /RDF/           ;READ FORWARD.
2680 003740    122     104     122             .ASCII  /RDR/           ;READ REVERSE.
2681 003743    127     122     124             .ASCII  /WRT/           ;WRITE
2682 003746    127     124     126             .ASCII  /WTV/           ;WRITE/VERIFY.  (WRITE ALL RECORDS, RDR AND CHECK DATA
2683                                                                    ;ON ALL RECORDS, RDF AND CHECK DATA ON ALL RECORDS.)
2684 003751    123     122     106             .ASCII  /SRF/           ;SPACE "N" RECORDS FORWARD.
2685 003754    123     122     122             .ASCII  /SRR/           ;SPACE "N"  RECORDS REVERSE.
2686 003757    122     116     122             .ASCII  /RNR/           ;READ NEXT REVERSE. I.E., SPACE FWD READ REVERSE.
2687 003762    122     116     106             .ASCII  /RNF/           ;READ NEXT FORWARD, I.E., READ FORWARD, SPACE REVERSE.
2688 003765    122     120     106             .ASCII  /RPF/           ;READ PREVIOUS FORWARD. IE., SPACE REVERSE, READ FORWARD
2689 003770    122     120     122             .ASCII  /RPR/           ;READ PREVIOUS REVERSE. IE., READ REVERSE, SPACE FORWARD
2690 003773    127     122     122             .ASCII  /WRR/           ;WRITE RETRY.
2691 003776    122     127     104             .ASCII  /RWD/           ;REWIND.
2692 004001    115     102     122             .ASCII  /MBR/           ;MESSAGE BUFFER RELEASE
2693 004004    127     124     115             .ASCII  /WTM/           ;WRITE TAPE MARK
2694 004007    127     124     122             .ASCII  /WTR/           ;WRITE TAPE MARK RETRY.
2695 004012    123     106     106             .ASCII  /SFF/           ;SPACE "N" FILES FORWARD.
2696 004015    123     106     122             .ASCII  /SFR/           ;SPACE "N" FILES REVERSE.
2697 004020    107     105     123             .ASCII  /GES/           ;GET EXTENDED STATUS.
2698 004023    105     122     123             .ASCII  /ERS/           ;ERASE 3 INCHES OF TAPE.
2699 004026    125     116     114             .ASCII  /UNL/           ;REWIND AND UNLOAD.
2700 004031    103     114     116             .ASCII  /CLN/           ;CLEAN TAPE.
2701 004034    123     103     110             .ASCII  /SCH/           ;SET CHARACTERISTICS.  WHERE BRF=200, 40, 20, 0.
2702                                                                    ;SEE TK25 PROGRAMMING SPECIFICATION FOR DESCRIPTION.
2703 004037    104     111     101             .ASCII  /DIA/           ;DIAGNOSTICS.  SEE TK25 PROGRAMMING SPECIFICATION
2704                                                                    ;FOR DESCRIPTION.  ODT MUST BE USED TO LOAD DIAGNOSTIC DATA
2705                                                                    ;INTO THE WRITE BUFFER BEFORE THIS CMD IS ISSUED.
2706 004042    112     115     120             .ASCII  /JMP/           ;JUMP TO THE NTH COMMAND IN THE COMMAND
2707                                                                    ;SEQUENCE TABLE, WHERE N IS DEFINED IN
2708                                                                    ;THE # OF OPERATIONS.
2709 004045    104     114     131             .ASCII  /DLY/           ;DELAY "N" MS, WHERE N IS DEFINED IN
2710                                                                    ;THE # OF OPERATIONS.
2711 004050    105     116     104             .ASCII  /END/           ;END OF COMMAND SEQUENCE.
2712                                            .EVEN
2713
2714
2715
```

```
                                .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

004054  045  116  045  CODELM:: .ASCIZ /%N%AUNIT %D1%A TK25 CODE LEVEL P%03%N%N/
                                .EVEN
004124  130  130  130  HALTM:: .ASCIZ  /XXX CMD - TYPE <CR> TO CONTINUE/
004164  103  115  104  CMDPKM:: .ASCIZ /CMD PACKET ADR NOT ON MODULO 4 BOUNDARY; RELOAD!/
                                .EVEN
004246  104  101  124  WTVERM:: .ASCIZ /DATA COMPARE ERROR/
004271  116  117  040  TOERM:: .ASCIZ  /NO TK25 RESPONSE/
004312  125  116  104  SCERM:: .ASCIZ  /UNDEFINED SPEC COND/
004336  122  106  103  RFCERM:: .ASCIZ /RFC NON ZERO/
004353  124  113  062  NSSRM:: .ASCIZ  /TK25 NOT READY/
004372  122  105  124  RLEXM:: .ASCIZ  /RETRY LIMIT EXCEEDED/
004417  125  116  111  ATTNM:: .ASCIZ  /UNIT OFF LINE/
004435  106  125  116  FUNRM:: .ASCIZ  /FUNCTION REJECT/
004455  106  101  124  FATSM:: .ASCIZ  /FATAL SUBSYSTEM ERROR/
004503  116  117  040  NOINTM:: .ASCIZ /NO INTERRUPT/
004520  124  101  120  TSAM::  .ASCIZ  /TAPE STATUS ALERT/
004542  124  117  117  TOOMM:: .ASCIZ  /TOO MANY INTERRUPTS/
004566  103  101  120  RNYM::  .ASCIZ  /CAPSTAN RUNAWAY-GET STATUS RESULTS:/
004632  122  105  103  RERM::  .ASCIZ  /RECOVERABLE ERROR/
004654  125  116  122  URERM:: .ASCIZ  /UNRECOVERABLE ERROR/
004700  045  116  045  DROPDM:: .ASCIZ /%N%ADROPPED UNIT %D1%N/
004727  045  116  045  AUDRPM:: .ASCIZ /%N%AALL UNITS DROPPED%N%N/
004761  045  116  045  DTAER2:: .ASCIZ "%N%ABYTE:%D4%S2%AWAS:%B8%S2%AS/B:%B8%N"
005030  045  104  064  DTAER3:: .ASCIZ "%D4%A BYTES IN ERROR OUT OF %D4%N"
005072  045  101  116  DTAER4:: .ASCIZ /%ANO DATA READ%N/
005113  045  101  122  DTAER5:: .ASCIZ /%ARECORD TOO LONG: >%O4%A BYTES%N/
005155  045  101  122  NURTY1:: .ASCIZ /%ARECOVERED ON RETRY #%D2%N/
005211  045  101  125  OFLINM:: .ASCIZ /%AUNIT %D1%A OFF LINE%N/
005241  045  101  107  GETSTM:: .ASCIZ /%AGET STATUS CMD RESULTS:%N/
005275  045  116  000  CRLF::  .ASCIZ  /%N/
005300  045  116  045  CRLFSP:: .ASCIZ /%N%S7/
005306  045  116  045  RAMFHR: .ASCIZ  '%N%A ********** CONTROLLER RAM DUMP **********'
005365  045  116  045  RAMIOP: .ASCIZ  '%N%A RAM ADDRESS (OCTAL) = %O3%A - %O3%N'
005436  045  101  040  RAMPD:  .ASCIZ  '%A %O3%A '
005450  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..
                                ;--


                                        BGNMSG  DTAERM
                                DTAERM::
                                        PRINTB  #STAER1,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
005460  016546  003330                                                    MOV     RECCNT(R5),-(SP)
005464  016546  003260                                                    MOV     PASCNT(R5),-(SP)
005470  016546  002536                                                    MOV     DEVTBL(R5),-(SP)
005474  012746  006140                                                    MOV     #STAER1,-(SP)
005500  012746  000004                                                    MOV     #4,-(SP)
005504  010600                                                            MOV     SP,R0
005506  104414                                                            TRAP    C$PNTB
005510  062706  000012                                                    ADD     #12,SP
                                        PRINTB  #STAER7
005514  012746  006232                                                    MOV     #STAER7,-(SP)
005520  012746  000001                                                    MOV     #1,-(SP)
005524  010600                                                            MOV     SP,R0
005526  104414                                                            TRAP    C$PNTB
005530  062706  000004                                                    ADD     #4,SP
                                        LET RECRED :* R2                ;SAVE R2
005534  010237  006546                                                            MOV     R2,RECRED
                                        LET TIME1 :* R3                 ;SAVE R3
005540  010337  003444                                                            MOV     R3,TIME1
                                        LET TIME2 :* R4                 ;SAVE R4
005544  010437  003446                                                            MOV     R4,TIME2
005550  004737  006602                JSR PC,RECTAP                   ;RETRIEVE RECORD READ
                                        LET R2 :* RECRED                ;RESTORE R2
005554  013702  006546                                                            MOV     RECRED,R2
                                        LET RECRED :* R3                ;SAVE RECORD READ
005560  010337  006546                                                            MOV     R3,RECRED
                                        LET R3 :* TIME1                 ;RESTORE R3
005564  013703  003444                                                            MOV     TIME1,R3
                                        LET R4 :* TIME2                 ;RESTORE R4
005570  013704  003446                                                            MOV     TIME2,R4
                                        PRINTB  #STAER6,RECRED          ;PRINT RECORD READ
005574  013746  006546                                                    MOV     RECRED,-(SP)
005600  012746  006362                                                    MOV     #STAER6,-(SP)
005604  012746  000002                                                    MOV     #2,-(SP)
005610  010600                                                            MOV     SP,R0
005612  104414                                                            TRAP    C$PNTB
005614  062706  000006                                                    ADD     #6,SP
                                        EXIT    MSG
005620  000167                                                            .WORD   J$JMP
005622  000000                                                            .WORD   L10002-2 .

                                        .EVEN

                                        ENDMSG
                                L10002:
005624  104423                                                            TRAP    C$MSG
```

```
2808 005626                             BGNMSG  STAERM
     005626                       STAERM::
2809 005626                             PRINTB  #STAER1,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
     005626  016546  003330                                                  MOV     RECCNT(R5),-(SP)
     005632  016546  003260                                                  MOV     PASCNT(R5),-(SP)
     005636  016546  002536                                                  MOV     DEVTBL(R5),-(SP)
     005642  012746  006140                                                  MOV     #STAER1,-(SP)
     005646  012746  000004                                                  MOV     #4,-(SP)
     005652  010600                                                          MOV     SP,R0
     005654  104414                                                          TRAP    C$PNTB
     005656  062706  000012                                                  ADD     #12,SP
2810 005662                             PRINTB  #STAER7
     005662  012746  006232                                                  MOV     #STAER7,-(SP)
     005666  012746  000001                                                  MOV     #1,-(SP)
     005672  010600                                                          MOV     SP,R0
     005674  104414                                                          TRAP    C$PNTB
     005676  062706  000004                                                  ADD     #4,SP
2811 005702                             LET R2 := CMDPKT CLR.BY #177740
     005702  013702  002314                                                          MOV     CMDPKT,R2
     005706  042702  177740                                                          BIC     #177740,R2
2812 005712                             LET R2 := R2 - #1
     005712  005302                                                                  DEC     R2
2813 005714                             IF R2 EQ #0 THEN                ;IF CMD IS A READ
     005714  005702                                                                  TST     R2
     005716  001016                                                                  BNE     50000$
2814 005720  004737  006602               JSR PC,RECTAP                 ;THEN RETRIEVE
2815 005724                               LET RECRED := R3              ;AND
     005724  010337  006546                                                          MOV     R3,RECRED
2816 005730                               PRINTB #STAER6,RECRED         ;TYPE RECORD READ
     005730  013746  006546                                                  MOV     RECRED,-(SP)
     005734  012746  006262                                                  MOV     #STAER6,-(SP)
     005740  012746  000002                                                  MOV     #2,-(SP)
     005744  010600                                                          MOV     SP,R0
     005746  104414                                                          TRAP    C$PNTB
     005750  062706  000006                                                  ADD     #6,SP
2817 005754                             ENDIF
     005754                                                          50000$:
2818 005754                             PRINTX  #STAER2
     005754  012746  006316                                                  MOV     #STAER2,-(SP)
     005760  012746  000001                                                  MOV     #1,-(SP)
     005764  010600                                                          MOV     SP,R0
     005766  104415                                                          TRAP    C$PNTX
     005770  062706  000004                                                  ADD     #4,SP
2819 005774                             PRINTX  #STAER3,CMDPKT,@TSDB(R5),MSGPKT+MS.RFC,TSSREG,CTCC
     005774  013746  003456                                                  MOV     CTCC,-(SP)
     006000  013746  003462                                                  MOV     TSSREG,-(SP)
     006004  013746  002344                                                  MOV     MSGPKT+MS.RFC,-(SP)
     006010  017546  002456                                                  MOV     @TSDB(R5),-(SP)
     006014  013746  002314                                                  MOV     CMDPKT,-(SP)
     006020  012746  006375                                                  MOV     #STAER3,-(SP)
     006024  012746  000006                                                  MOV     #6,-(SP)
     006030  010600                                                          MOV     SP,R0
     006032  104415                                                          TRAP    C$PNTX
     006034  062706  000016                                                  ADD     #16,SP
2820 006040                             PRINTX  #STAER4,CMDPKT+2,CMDPKT+4,CMDPKT+6
     006040  013746  002322                                                  MOV     CMDPKT+6,-(SP)
     006044  013746  002320                                                  MOV     CMDPKT+4,-(SP)
```

```
        006050  013746  002316                                                          MOV     CMDPKT+2,-(SP)
        006054  012746  006433                                                          MOV     #STAER4,-(SP)
        006060  012746  000004                                                          MOV     #4,-(SP)
        006064  010600                                                                  MOV     SP,R0
        006066  104415                                                                  TRAP    C$PNTX
        006070  062706  000012                                                          ADD     #12,SP
  2821  006074                          PRINTX  #STAER5,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3
        006074  013746  002354                                                          MOV     MSGPKT+MS.XS3,-(SP)
        006100  013746  002352                                                          MOV     MSGPKT+MS.XS2,-(SP)
        006104  013746  002350                                                          MOV     MSGPKT+MS.XS1,-(SP)
        006110  013746  002346                                                          MOV     MSGPKT+MS.XS0,-(SP)
        006114  012746  006453                                                          MOV     #STAER5,-(SP)
        006120  012746  000005                                                          MOV     #5,-(SP)
        006124  010600                                                                  MOV     SP,R0
        006126  104415                                                                  TRAP    C$PNTX
        006130  062706  000014                                                          ADD     #14,SP
  2822  006134                          EXIT    MSG
        006134  000167                                                                 .WORD   J$JMP
        006136  000410                                                                 .WORD   L10003-2-.
  2823
  2824                                  .NLIST  BEX
  2825  006140     045     101     130  STAER1: .ASCIZ  /%AXXX CMD FAILED - UNIT %D1%S3%APASS:%D5%S3%ARECORD:%D5%N/
  2826                                  .EVEN
  2827  006232     045     101     120  STAER7: .ASCIZ  /%APREVIOUS CMD WAS  XXX/
  2828  006262     045     123     061  STAER6: .ASCIZ  /%S11%A* RECORD READ:%D5%A */
  2829  006316     045     116     045  STAER2: .ASCIZ  /%N%ACMDPKT%S2%ATSBA* 4%ARFC%S5%ATSSR%S3%ATCC%N/
  2830  006375     045     117     066  STAER3: .ASCIZ  /%06%S2%06%S2%06%S2%06%S2%D1%N/
  2831  006433     045     117     066  STAER4: .ASCII  /%06%N/
  2832  006440     045     117     066          .ASCII  /%06%N/
  2833  006445     045     117     066          .ASCIZ  /%06%N/
  2834  006453     045     101     130  STAER5: .ASCII  /%AXST0%S4%AXST1%S4%AXST2%S4%AXST3%N/
  2835  006516     045     117     066          .ASCIZ  /%06%S2%06%S2%06%S2%06%N/
  2836                                  .LIST   BEX
  2837                                  .EVEN
  2838  006546  000000                  RECRED: .WORD   0                       ;RECORD READ FROM TAPE
  2839
  2840  006550                          ENDMSG
        006550                          L10003:
        006550  104423                                                                  TRAP    C$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
                                L10004:
                                                                                RTI

                                        BGNSRV  TS4IN1                  ;DEVICE 1.
                                TS4IN1::
                                        LET INTFLG+2 := INTFLG+2 + #1    ;SET INTERRUPT OCCURRED FLAG.
                                                                                INC     INTFLG+2
                                        ENDSRV
                                L10005:
                                                                                RTI

                                        BGNSRV  TS4IN2                  ;DEVICE 2.
                                TS4IN2::
                                        LET INTFLG+4 := INTFLG+4 + #1    ;SET INTERRUPT OCCURRED FLAG.
                                                                                INC     INTFLG+4
                                        ENDSRV
                                L10006:
                                                                                RTI

                                        BGNSRV  TS4IN3                  ;DEVICE 3.
                                TS4IN3::
                                        LET INTFLG+6 := INTFLG+6 + #1    ;SET INTERRUPT OCCURRED FLAG.
                                                                                INC     INTFLG+6
                                        ENDSRV
                                L10007:
                                                                                RTI
```

```
2866
2857                                    ;       SUBROUTINE TO RETRIEVE RECORD COUNT READ FROM TAPE FOR ERROR
2868                                    ;       PRINTS.
2869                                    ;       INPUTS:
2870                                    ;       OUTPUTS:  R3 = RECORD COUNT READ
2871                                    ;       REGISTERS:  R2, R3, R4
2872                                    ;       CALLS:
2873
2874 006602                            RECTAP::IF #MOD.C0 SETIN CMDWRD THEN    ;READ REV FETCH
     006602  032737  000400  003426                                                  BIT     #MOD.C0,CMDWRD
     006610  001430                                                                 BEQ     50001$
2875 006612                                      LET R2 := MSGPKT+MS.RFC + DATARD ;FIND LAST READ AD.
     006612  013702  002344                                                         MOV     MSGPKT+MS.RFC,R2
     006616  063702  003416                                                         ADD     DATARD,R2
2876 006622                                      IF #BIT00 SETIN R2 THEN        ;ODD AD., REASSEMBLE
     006622  032702  000001                                                         BIT     #BIT00,R2
     006626  001417                                                                 BEQ     50002$
2877 006630                                        LET R2 := R2 + #1            ;REC COUNT STARTING
     006630  005202                                                                 INC     R2
2878 006632                                        LET R3 :B= (R2) CLR.BY #177400  ;WITH UPPER BYTE FETCH
     006632  111203                                                                 MOVB    (R2),R3
     006634  142703  177400                                                         BICB    #177400,R3
2879 006640                                        LET R3 := SWAP R3            ;
     006640  000303                                                                 SWAB    R3
2880 006642                                        LET R2 := R2 - #1            ;LOWER BYTE AD.
     006642  005302                                                                 DEC     R2
2881 006644                                        IFB SWBFLG NE #0 THEN
     006644  105737  003524                                                         TSTB    SWBFLG
     006650  001401                                                                 BEQ     50003$
2882 006652                                          LET R2 := R2 - #1          ;LOWER BYTE AD. ON SWAP
     006652  005302                                                                 DEC     R2
2883 006654                                        ENDIF
     006654                                                                        50003$:
2884 006654                                        LET R4 :B= (R2) CLR.BY #177400 ;FETCH LOWER BYTE
     006654  111204                                                                 MOVB    (R2),R4
     006656  142704  177400                                                         BICB    #177400,R4
2885 006662                                        LET R3 := R3 OR R4           ;MERGE BYTES
     006662  050403                                                                 BIS     R4,R3
2886 006664                                      ELSE
     006664  000401                                                                 BR      50004$
     006666                                                                        50002$:
2887 006666                                        LET R3 := (R2)               ;EVEN AD. FETCH
     006666  011203                                                                 MOV     (R2),R3
2888 006670                                      ENDIF
     006670                                                                        50004$:
2889 006670                                    ELSE
     006670  000402                                                                 BR      50005$
     006672                                                                        50001$:
2890 006672                                      LET R3 := @DATARD              ;READ FWD FETCH
     006672  017703  174520                                                         MOV     @DATARD,R3
2891 006676                                    ENDIF
     006676                                                                        50005$:
2892
2893 006676  000207                    RTS     PC
```

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

006700                          SETCH:: LET R1 := #CMDSEQ               ;INIT COMMAND SEQUENCE TABLE POINTER.
006700  012701  003542                                                          MOV     #CMDSEQ,R1
006704  012721  140004                  MOV     #SCH,(R1)+              ;THIS CODE SETS UP A SET CHARACTERISTIC
006710  012721  000040                  MOV     #DFTSCH,(R1)+           ;COMMAND AS THE FIRST COMMAND IN THE
006714  012721  000001                  MOV     #1,(R1)+                ;SEQUENCE TABLE.
006720  005721                          TST     (R1)+                   ;SKIP PATTERN LOCATION.
006722  000207                          RTS PC

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

006724                          SETRW:: LET (R1)+ := #RWD               ;CMD = REWIND.
006724  012721  102010                                                          MOV     #RWD,(R1)+
006730                                  LET (R1)+ := #1                 ;BRF.
006730  012721  000001                                                          MOV     #1,(R1)+
006734                                  LET (R1)+ := #1                 ;# OF OPERATIONS.
006734  012721  000001                                                          MOV     #1,(R1)+
006740  005721                          TST (R1)+                       ;SKIP PATTERN.
006742  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.

                                EXALL:: LET R1 := @CMDSEQ               ;INIT SEQUENCE TABLE POINTER.
006744  012701  003542                                                    MOV     @CMDSEQ,R1
                                        WHILE (R1) NE @END DO           ;WHILE THERE ARE CMDS IN THE SEQUENCE TABLE.
006750                                                                  50006$:
006750  021127  177777                                                    CMP     (R1),@END
006754  001527                                                            BEQ     50007$
                                          JSR   PC,SETUP                ;GO SETUP THE COMMAND BLOCK.
                                          WHILE NCNT LT NCNT1 DO        ;WHILE THERE ARE RECORDS REMAINING:
006762                                                                  50010$:
006762  023737  003420  003422                                          CMP     NCNT,NCNT1
006770  002116                                                          BGE     50011$
                                            JSR PC,CMDAC                ;STORE CMD ASCII IN ERROR MESSAGE.
                                            IFB RANDOM NE @0 THEN       ;IF IN RANDOM MODE:
006776  105737  003521                                                    TSTB    RANDOM
007002  001435                                                            BEQ     50012$
                                              IF CMDWRD EQ @WRT THEN    ;IF CMD IS A WRITE THEN:
007004  023727  003426  104005                                          CMP     CMDWRD,@WRT
007012  001031                                                            BNE     50013$
                                                IFB VFYFLG EQ @0 THEN   ;IF DATA IS NOT TO BE VERIFIED THEN:
007014  105737  003522                                                    TSTB    VFYFLG
007020  001026                                                            BNE     50014$
                                                  LET RANB := RANB + RANS  ;GENERATE
007022  063737  003442  003440                                          ADD     RANS,RANB
                                                LET RANS := RANS + RANB  ;RANDOM
007030  063737  003440  003442                                          ADD     RANB,RANS
                                                LET BRFCNT := RANS       ;LENGTH
007036  013737  003442  003424                                          MOV     RANS,BRFCNT
                                                LET BRFCNT := BRFCNT CLR.BY LENMSK  ;MASK RANDOM LENGTH.
007044  043737  003436  003424                                          BIC     LENMSK,BRFCNT
                                                IF BRFCNT LT @18. THEN          ;DO NOT ALLOW BYTE COUNT OF LESS THAN 18.
007052  023727  003424  000022                                          CMP     BRFCNT,@18.
007060  002003                                                            BGE     50015$
                                                  LET BRFCNT := @18.    ;CHANGE COUNT OF 0-17 TO 18.
007062  012737  000022  003424                                          MOV     @18.,BRFCNT
                                                ENDIF
007070                                                                  50015$:
                                                LET CMDPKT+CP.CNT := BRFCNT  ;MOVE BRF TO CMD PACKET.
007070  013737  003424  002322                                          MOV     BRFCNT,CMDPKT+CP.CNT
                                              ENDIF
007076                                                                  50014$:
                                            ENDIF
007076                                                                  50013$:
                                          ENDIF
007076                                                                  50012$:
007076  004737  007240                  JSR PC,EXSUB                    ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATUS.
007102  004737  017706                  JSR PC,CKHAE                    ;CHECK HALT AFTER EACH CMD FLAG.
007106                                  LET R2 := @1                    ;SET ALL UNITS AT BOT/EOT.
007106  012702  000001                                                    MOV     @1,R2
007112  004737  017300                  JSR PC,FIRSTU                   ;FIND FIRST UNIT.
007116                                  WHILE DEVTBL(R5) NE @END DO     ;WHILE THERE ARE MORE UNITS:
```

```
                007116
                007116  026527  002536  177777
                007124  001426
          2956  007126
                007126  032737  000400  003426
                007134  001406
          2957  007136
                007136  032765  000002  003506
                007144  001001
          2958  007146
                007146  005002
          2959  007150
                007150
          2960  007150
                007150  000411
                007152
          2961  007152
                007152  032765  000001  003506
                007160  001404
                007162  032737  000001  003426
                007170  001001
                007172
          2962
          2963  007172
                007172  005002
          2964  007174
                007174
          2965  007174
                007174
          2966  007174  004737  017346
          2967  007200
                007200  000746
                007202
          2968  007202
                007202  020227  000001
                007206  001001
          2969  007210  000412
          2970  007212
                007212
          2971  007212
                007212  005237  003420
          2972  007216
                007216  013737  003426  003432
          2973  007224
                007224  000656
                007226
          2974  007226  004737  016142
          2975
          2976  007232
                007232  000646
                007234
          2977  007234
                007234  005002
          2978  007236  000207
          2979
          2980
          2981
```

```
                                                50016$:
                                                CMP     DEVTBL(R5),#END
                                                BEQ     50017$
IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
                                                BIT     #MOD.CO,CMDWRD
                                                BEQ     50020$
  IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT THEN:
                                                BIT     #XO.BOT,EOTFLG(R5)
                                                BNE     50021$
    LET R2 := #0                                ;CLEAR EOT/BOT FLAG.
                                                CLR     R2
  ENDIF
                                                50021$:
ELSE                            ;ELSE IF CMD IS NOT REVERSE:
                                                BR      50022$
                                                50020$:
  IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
                                                BIT     #XO.EOT,EOTFLG(R5)
                                                BEQ     50023$
                                                BIT     #CMD.CO,CMDWRD
                                                BNE     50024$
                                                50023$:
                                ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
    LET R2 := #0                  ;CLEAR EOT/BOT FLAG.
                                                CLR     R2
  ENDIF
                                                50024$:
ENDIF
                                                50022$:
    JSR PC,NEXTU                ;FIND NEXT UNIT
  ENDDO                         ;
                                                BR      50016$
                                                50017$:
  IF R2 EQ #1 THEN              ;IF ALL UNIT ARE AT EOT/BOT THEN:
                                                CMP     R2,#1
                                                BNE     50025$
    BR          EXARTN          ;RETURN WITH R2 = #1.
  ENDIF
                                                50025$:
  LET NCNT := NCNT + #1         ;UPDATE RECORD COUNT.
                                                INC     NCNT
  LET PCMDWD := CMDWRD          ;SAVE PREVIOUS COMMAND WORD.
                                                MOV     CMDWRD,PCMDWD
ENDDO
                                                BR      50010$
                                                50011$:
  JSR   PC,VFYDAT               ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
                                ;VERIFY THE LAST N RECORDS OF DATA.
ENDDO
                                                BR      50006$
                                                50007$:
  LET R2 := #0                  ;SET NORMAL RETURN INDICATOR.
                                                CLR     R2
EXARTN: RTS PC                  ;RETURN.
```

```
2982                                    ;       SUBROUTINE TO ISSUE COMMAND TO ALL DEVICES, WAIT FOR
2983                                    ;       ALL INTERRUPTS, AND CHECK ALL STATUS.
2984                                    ;       INPUTS:
2985                                    ;       OUTPUTS:
2986                                    ;       REGISTERS:
2987                                    ;       CALLS:          EXCUTE,GOWAIT,NEXTU,FIRSTU.
2988
2989 007240  004737  017300            EXSUB::     JSR PC,FIRSTU               ;SET UP FOR FIRST UNIT.
2990 007244                                         WHILE DEVTBL(R5) NE @END DO ;WHILE THERE ARE MORE DEVICES:
      007244                                                                        50026$:
      007244  026527  002536  177777                                              CMP     DEVTBL(R5),@END
      007252  001465                                                              BEQ     50027$
2991 007254                                           IF @MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
      007254  032737  000400  003426                                              BIT     @MOD.CO,CMDWRD
      007262  001421                                                              BEQ     50030$
2992 007264                                             IF @XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
      007264  032765  000002  003506                                              BIT     @XO.BOT,EOTFLG(R5)
      007272  001014                                                              BNE     50031$
2993 007274                                               IF @XO.EOT SETIN EOTFLG(R5) THEN  ;BUT IF AT EOT
      007274  032765  000001  003506                                              BIT     @XO.EOT,EOTFLG(R5)
      007302  001406                                                              BEQ     50032$
2994 007304                                                 IFB ALLEOT NE @0 THEN       ;AND ALL OTHERS AT EOT
      007304  105737  003531                                                       TSTB    ALLEOT
      007310  001402                                                              BEQ     50033$
2995 007312  004737  010654                                    JSR PC,EXCUTE            ;THEN EXECUTE REV CMD
2996 007316                                                   ENDIF                     ;IF NOT ALL AT EOT, FREEZE UNIT(S) AT EOT
      007316                                                                        50033$:
2997 007316                                                 ELSE                      ;IF NOT AT BOT AND
      007316  000402                                                              BR      50034$
      007320                                                                        50032$:
2998 007320  004737  010654                                    JSR PC,EXCUTE            ;NOT AT EOT, EXEC REV CMD
2999 007324                                                   ENDIF
      007324                                                                        50034$:
3000 007324                                               ENDIF
      007324                                                                        50031$:
3001 007324                                           ELSE                        ;ELSE IF CMD IS NOT REVERSE:
      007324  000435                                                              BR      50035$
      007326                                                                        50030$:
3002 007326                                             IF CMDLG EQ @2 AND @XO.BOT SETIN EOTFLG(R5) THEN
      007326  023727  003434  000002                                              CMP     CMDLG,@2
      007334  001011                                                             BNE     50036$
      007336  032765  000002  003506                                              BIT     @XO.BOT,EOTFLG(R5)
      007344  001405                                                             BEQ     50036$
3003                                                                            ;CLEAR BAD SPOT COUNTS WHEN WRITING FROM BOT
3004 007346                                               LET BTPT := BTADDR(R5)
      007346  016537  002550  003516                                              MOV     BTADDR(R5),BTPT
3005 007354                                               LET @BTPT := @0
      007354  005077  174136                                                      CLR     @BTPT
3006 007360                                             ENDIF
      007360                                                                        50036$:
3007 007360                                             IF @XO.EOT NOTSETIN EOTFLG(R5) OR @CMD.CO NOTSETIN CMDWRD THEN
      007360  032765  000001  003506                                              BIT     @XO.EOT,EOTFLG(R5)
      007366  001404                                                             BEQ     50037$
      007370  032737  000001  003426                                              BIT     @CMD.CO,CMDWRD
      007376  001003                                                             BNE     50040$
      007400                                                                        50037$:
3008                                                                            ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
```

```
3009 007400  004737  010654                    JSR PC,EXCUTE           ;ISSUE CMD TO TK25.
3010 007404                                   ELSE
     007404  000405                                                     BR      50041$
     007406                                                             50040$:
3011 007406                                     IFB ALLEOT NE #0 THEN
     007406  105737  003531                                            TSTB    ALLEOT
     007412  001402                                                     BEQ     50042$
3012 007414  004737  010654                      JSR PC,EXCUTE
3013 007420                                     ENDIF
     007420                                                             50042$:
3014 007420                                     ENDIF
     007420                                                             50041$:
3015 007420                                   ENDIF
     007420                                                             50035$:
3016 007420  004737  017346                   JSR       PC,NEXTU      ;FIND NEXT UNIT IN TEST CYCLE.
3017 007424                                   ENDDO
     007424  000707                                                     BR      50026$
     007426                                                             50027$:
3018 007426                                 IFB RPTFLG NE #0 THEN       ;IF REPORT HAS BEEN REQUESTED THEN:
     007426  105737  003523                                            TSTB    RPTFLG
     007432  001403                                                     BEQ     50043$
3019 007434                                   LET RPTFLG :B= #0         ;CLR THE FLAG.
     007434  105037  003523                                            CLRB    RPTFLG
3020 007440                                   DORPT                     ;PRINT THE PERFORMANCE REPORT.
     007440  104424                                                     TRAP    C$DRPT
3021 007442                                 ENDIF                       ;
     007442                                                             50043$:
3022 007442  004737  017300                 JSR PC,FIRSTU               ;SET UP FOR FIRST UNIT.
3023 007446                                 WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
     007446                                                             50044$:
     007446  026527  002536  177777                                     CMP     DEVTBL(R5),#END
     007454  001450                                                     BEQ     50045$
3024 007456                                   IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
     007456  032737  000400  003426                                     BIT     #MOD.CO,CMDWRD
     007464  001421                                                     BEQ     50046$
3025 007466                                     IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
     007466  032765  000002  003506                                     BIT     #XO.BOT,EOTFLG(R5)
     007474  001014                                                     BNE     50047$
3026 007476                                       IF #XO.EOT SETIN EOTFLG(R5) THEN   ;BUT IF AT EOT
     007476  032765  000001  003506                                     BIT     #XO.EOT,EOTFLG(R5)
     007504  001406                                                     BEQ     50050$
3027 007506                                         IFB ALLEOT NE #0 THEN           ;AND ALL OTHERS AT EOT
     007506  105737  003531                                            TSTB    ALLEOT
     007512  001402                                                     BEQ     50051$
3028 007514  004737  011274                          JSR PC,GOWAIT                  ;THEN WAIT FOR CMD END
3029 007520                                         ENDIF                           ;IF NOT ALL AT EOT, DO NOT WAIT
     007520                                                             50051$:
3030 007520                                       ELSE                              ;NOT AT BOT, AND NOT AT EOT
     007520  000402                                                     BR      50052$
     007522                                                             50050$:
3031 007522  004737  011274                        JSR PC,GOWAIT                   ;WAIT FOR INT,CHECK STATUS.
3032 007526                                       ENDIF
     007526                                                             50052$:
3033 007526                                     ENDIF
     007526                                                             50047$:
3034 007526                                   ELSE                      ;ELSE IF CMD IS FORWARD:
     007526  000420                                                     BR      50053$
```

F6


```
                007530
3035            007530
                007530  032765  000001  003506          IF @XO.EOT NOTSETIN EOTFLG(R5) OR @CMD.CO NOTSETIN CMDWRD THEN
                007536  001404                                  BIT     @XO.EOT,EOTFLG(R5)
                007540  032737  000001  003426                  BEQ     50054$
                007546  001003                                  BIT     @CMD.CO,CMDWRD
                007550                                          BNE     50055$
3036                                                            50054$:
3037            007550  004737  011274                  ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
                                                          JSR PC,GOWAIT           ;WAIT FOR INT,CHECK STATUS.
3038            007554                                  ELSE
                007554  000405                                  BR      50056$
                007556                                          50055$:
3039            007556                                    IFB ALLEOT NE @0 THEN
                007556  105737  003531                          TSTB    ALLEOT
                007562  001402                                  BEQ     50057$
3040            007564  004737  011274                      JSR PC,GOWAIT
3041            007570                                    ENDIF
                007570                                          50057$:
3042            007570                                  ENDIF
                007570                                          50056$:
3043            007570                                ENDIF
                007570                                          50053$:
3044            007570  004737  017346                JSR       PC,NEXTU        ;FIND NEXT UNIT IN TEST CYCLE.
3045            007574                                ENDDO
                007574  000724                                  BR      50044$
                007576                                          50045$:
3046            007576  000207                  RTS PC                          ;RETURN.
```

```
3048                                    ;       THIS SUBROUTINE STORES THE ASCII FOR THE CURRENT COMMAND AND PREVIOUS
3049                                    ;       COMMAND IN THE STANDARD ERROR MESSAGE.  ON ENTRY LOCATION CMDWRD
3050                                    ;       CONTAINS CURRENT CMD AND LOCATION PCMDWD CONTAINS PREVIOUS CMD.
3051                                    ;       INPUTS:
3052                                    ;       OUTPUTS:
3053                                    ;       REGISTERS:      R3, R4.
3054                                    ;       CALLS:          GCMDA
3055
3056 007600                             CMDAC:: LET R4 := CMDWRD                ;R4 = CMD BINARY.
     007600  013704  003426                                                             MOV     CMDWRD,R4
3057 007604  004737  007654                     JSR     PC,GCMDA                ;GET CMD ASCII.
3058 007610  112337  006142                     MOVB    (R3)+,STAER1+2          ;MOVE CMD ASCII
3059 007614  112337  006143                     MOVB    (R3)+,STAER1+3          ;
3060 007620  111337  006144                     MOVB    (R3),STAER1+4           ;INTO MSG.
3061 007624                                     LET R4 := PCMDWD                ;R4 = PREVIOUS CMD BINARY.
     007624  013704  003432                                                             MOV     PCMDWD,R4
3062 007630  004737  007654                     JSR     PC,GCMDA                ;GET CMD ASCII.
3063 007634  000240                             NOP
3064 007636                                     LET STAER7+24 :B= (R3)+         ;MOVE CMD ASCII
     007636  112337  006256                                                             MOVB    (R3)+,STAER7+24
3065 007642                                     LET STAER7+25 :B= (R3)+         ;
     007642  112337  006257                                                             MOVB    (R3)+,STAER7+25
3066 007646                                     LET STAER7+26 :B= (R3)          ;INTO MSG.
     007646  111337  006260                                                             MOVB    (R3),STAER7+26
3067 007652  000207                             RTS     PC                      ;RETURN. GO EXECUTE NEXT FUNCTION.
3068
3069
3070
3071                                    ;       SUBROUTINE TO FIND THE ASCII EQUIVILENT OF THE COMMAND IN R4.
3072                                    ;       ADDRESS OF ASCII 1ST WORD IS RETURNED IN R3.
3073                                    ;       INPUTS:         R4 = PRESENT COMMAND WORD.
3074                                    ;       OUTPUTS:        R3 = ADDRESS OF PRESENT COMMANO ASCII.
3075                                    ;       REGISTERS:
3076                                    ;       CALLS:
3076
3077
3078 007654                             GCMDA:: LET R3 := #0                    ;INIT CMD TBL POINTER.
     007654  005003                                                                     CLR     R3
3079 007656                                     WHILE CMDTBL(R3) NE R4 DO       ;UNTIL CURRENT CMD IS FOUND:
     007656                                                                             50060$:
     007656  026304  003644                                                             CMP     CMDTBL(R3),R4
     007662  001403                                                                     BEQ     50061$
3080 007664                                       LET R3 := R3 + #2             ;SEARCH CMD TABLE.
     007664  062703  000002                                                             ADD     #2,R3
3081 007670                                     ENDDO
     007670  000772                                                                     BR      50060$
     007672                                                                             50061$:
3082 007672                                     LET R4 := R3
     007672  010304                                                                     MOV     R3,R4
3083 007674                                     LET R3 := R3 SHIFT -1           ;POINT TO ASCII FOR THAT COMMAND
     007674  006203                                                                     ASR     R3
3084 007676  060403                             ADD     R4,R3
3085 007700  062703  003732                     ADD     #CMDASC,R3
3086 007704  000207                             RTS     PC                      ;RETURN.
```

```
3088                                    ;       THIS SUBROUTINE LOADS THE TK25 COMMAND PACKET FROM ONE
3089                                    ;       ENTRY IN THE SEQUENCE TABLE.
3090                                    ;       INPUTS:
3091                                    ;       OUTPUTS:
3092                                    ;       REGISTERS:      R2, R3.
3093                                    ;       CALLS:          GENPAT.
3094
3095 007706                            SETUP:: LET CMDLG := #0                 ;CLR CMD LOGGING CODE(DISABLES LOGGING)
     007706  005037  003434                                                             CLR     CMDLG
3096 007712  012137  002314                    MOV     (R1)+,CMDPKT            ;LOAD THE COMMAND WORD.
3097 007716  011137  002322                    MOV     (R1),CMDPKT+CP.CNT      ;LOAD THE BYTE/RECORD/FILE COUNT.
3098 007722  011137  003424                    MOV     (R1),BRFCNT             ;SAVE BRF FOR THIS COMMAND.
3099 007726  013702  002314                    MOV     CMDPKT,R2               ;GET CMD.
3100 007732  042702  177740                    BIC     #NCMD.C,R2              ;CLR ALL BUT CMD BITS.
3101 007736  010203                            MOV     R2,R3                   ;SAVE IT TWICE.
3102 007740  162703  000010                    SUB     #CMD.C3,R3              ;POSITION COMMAND?
3103 007744  001003                            BNE     2$                      ;BR IF NOT.
3104 007746  011137  002316                    MOV     (R1),CMDPKT+2           ;MOVE BPCR IN 2ND PKT WORD FOR POSITION CMD.
3105 007752  000461                            BR      3$
3106 007754                            2$:     IF CMDPKT EQ #WTM THEN          ;IF CMD IS A WRITE TAPE MARK THEN:
     007754  023727  002314  100011                                                     CMP     CMDPKT,#WTM
     007762  001003                                                                    BNE     50062$
3107 007764                                       LET CMDLG := #2                   ;WTM LOGGING CODE IS 2.
     007764  012737  000002  003434                                                     MOV     #2,CMDLG
3108 007772                                    ENDIF                                50062$:
     007772
3109 007772  010203                            MOV     R2,R3
3110 007774  162703  000001                    SUB     #CMD.C0,R3              ;IS IT A READ?
3111 010000  001017                            BNE     1$                      ;BR IF NOT.
3112 010002  013737  003416  002316            MOV     DATARD,CMDPKT+CP.ADL    ;IF SO, LOAD THE BUFFER ADDR.
3113 010010                            IF #MOD.C0 SETIN CMDPKT THEN    ;IF CMD IS A READ REV THEN:
     010010  032737  000400  002314                                                     BIT     #MOD.C0,CMDPKT
     010016  001404                                                                    BEQ     50063$
3114 010020                                       LET CMDLG := #4                   ;LOGGING CODE IS 4.
     010020  012737  000004  003434                                                     MOV     #4,CMDLG
3115 010026                            ELSE                                 ;ELSE - IF CMD IS A READ FWD:
     010026  000403                                                                    BR      50064$
     010030                                                                            50063$:
3116 010030                                       LET CMDLG := #6                   ;LOGGING CODE IS 6.
     010030  012737  000006  003434                                                     MOV     #6,CMDLG
3117 010036                            ENDIF
     010036                                                                            50064$:
3118 010036  000427                            BR      3$                      ;CONTINUE.
3119 010040  010203                    1$:     MOV     R2,R3                   ;IS IT
3120 010042  162703  000004                    SUB     #CMD.C2,R3              ;A SET CHARACTERISTICS CMD?
3121 010046  001011                            BNE     4$                      ;BR IF NOT.
3122 010050                            LET CMDPKT+CP.ADL := #SCHBK             ;SET UP ADR LO FOR SET CHAR.
     010050  012737  002446  002316                                                     MOV     #SCHBK,CMDPKT+CP.ADL
3123 010056  012737  000010  002322            MOV     #SCHCNT,CMDPKT+CP.CNT   ;SET BUFFER EXTENT
3124 010064                            LET SCHBK+6 := (R1)                     ;STORE CHARACTERISTIC CODE IN SCH BLOCK.
     010064  011137  002454                                                             MOV     (R1),SCHBK+6
3125 010070  000412                            BR      3$                      ;CONTINUE.
3126 010072  010203                    4$:     MOV     R2,R3                   ;IS IT
3127 010074  162703  000006                    SUB     #CMD.C1!CMD.C2,R3       ;A DIAGNOSTIC (DIA) CMD?
3128 010100  001006                            BNE     3$                      ;BR IF NOT.
3129 010102  012737  000020  002322            MOV     #DIACNT,CMDPKT+CP.CNT   ;LOAD BUFFER EXTENT.
3130 010110  012737  003414  002316            MOV     #DIABLK,CMDPKT+CP.ADL   ;LOAD BUFFER ADR LOW.
```

```
3131 010116  005721                    3$:    TST     (R1)+                ;POINT TO N (NUMBER OF TIMES TO EXECUTE THIS INSTRUC
3132 010120                                   LET NCNT1 := (R1)+             ;SAVE NUMBER OF OPERATIONS
     010120  012137  003422                                                  MOV     (R1)+,NCNT1
3133 010124                                   LET NCNT := #0                 ;CLEAR OPERATION COUNTER.
     010124  005037  003420                                                  CLR     NCNT
3134 010130  012137  003454                   MOV     (R1)+,PATERN           ;SAVE PATTERN CODE FOR CURRENT CMD.
3135 010134  010203                           MOV     R2,R3                  ;IS IT
3136 010136  162703  000005                   SUB     #CMD.CO!CMD.C2,R3      ;A WRITE?
3137 010142  001010                           BNE     5$                     ;BR IF NOT.
3138 010144  013737  003414  002316           MOV     DATAWT,CMDPKT+CP.ADL   ;LOAD WRITE BUFFER LO ORDER.
3139 010152  004737  010264                   JSR     PC,GENPAT              ;GO GENERATE THE WRITE PATTERN.
3140 010156                                   LET CMDLG := #2                ;WRITE LOGGING CODE IS 2.
     010156  012737  000002  003434                                          MOV     #2,CMDLG
3141 010164                           5$:     IF #VFY.C SETIN CMDPKT THEN    ;IF DATA VERIFICATION IS REQUIRED:
     010164  032737  000100  002314                                          BIT     #VFY.C,CMDPKT
     010172  001407                                                         BEQ     50065$
3142 010174                                     LET VFYFLG :B= #1            ;SET VERIFY FLAG.
     010174  112737  000001  003522                                          MOVB    #1,VFYFLG
3143 010202  042737  000100  002314              BIC   #VFY.C,CMDPKT        ;CLEAR VERIFY BIT(NOT USED BY HARDWARE).
3144 010210                                   ELSE                           ;IF DATA VERIFICATION IS NOT REQUIRED:
     010210  000402                                                         BR      50066$
     010212                                                                 50065$:
3145 010212                                     LET VFYFLG :B= #0            ;CLR VERIFY FLAG.
     010212  105037  003522                                                 CLRB    VFYFLG
3146 010216                                   ENDIF
     010216                                                                 50066$:
3147 010216                                   LET PCMDWD := CMDWRD           ;SAVE PREVIOUS CMD WORD.
     010216  013737  003426  003432                                          MOV     CMDWRD,PCMDWD
3148 010224                                   LET CMDWRD := CMDPKT           ;SAVE PRESENT CMD WORD.
     010224  013737  002314  003426                                          MOV     CMDPKT,CMDWRD
3149 010232                                   IFB SWBFLG NE #0 THEN          ;IF SWAP BYTES IS ENABLED:
     010232  105737  003524                                                 TSTB    SWBFLG
     010236  001403                                                         BEQ     50067$
3150 010240                                     LET CMDPKT := CMDPKT SET.BY #SWB.C  ;SET SWAP BIT IN COMMAND.
     010240  052737  010000  002314                                          BIS     #SWB.C,CMDPKT
3151 010246                                   ENDIF
     010246                                                                 50067$:
3152 010246  042737  004000  002314          BIC     #BRF.C,CMDPKT          ;CLR BRF BIT (INTERNAL ONLY).
3153 010254                                   LET CMDSAV := CMDPKT           ;SAVE 1ST WORD OF COMMAND PACKET.
     010254  013737  002314  003430                                          MOV     CMDPKT,CMDSAV
3154 010262  000207                           RTS     PC                     ;RETURN.
```

```
3156                                    ;       THIS SUBROUTINE SETS UP AND CALLS THE APPROPRIATE SUBROUTINE TO GENERATE
3157                                    ;       THE DESIRED PATTERN FOR THE WRITE AND WRITE/VERIFY COMMANDS.
3158                                    ;       INPUTS:
3159                                    ;       OUTPUTS:
3160                                    ;       REGISTERS:      R2, R3, R4.
3161                                    ;       CALLS:          PATR0 - PATR7
3162
3163 010264                            GENPAT:: LET R3 := PATERN SHIFT 1       ;SETUP PATTERN ROUTINE POINTER
     010264  013703  003454                                                     MOV     PATERN,R3
     010270  006303                                                             ASL     R3
3164 010272                                     LET R4 := BRFCNT + #1   ;SET LENGTH OF WRITE BFR
     010272  013704  003424                                                     MOV     BRFCNT,R4
     010276  005204                                                             INC     R4
3165 010300                                     LET R4 := R4 CLR.BY #1  ;ROUNDED UP TO NEXT WORD
     010300  042704  000001                                                     BIC     #1,R4
3166 010304                                     LET R4 := R4 - #2       ;WITH FIRST WORD RESERVED
     010304  162704  000002                                                     SUB     #2,R4
3167 010310                                     LET R2 := DATAWT + #2   ;FOR RECORD COUNT
     010310  013702  003414                                                     MOV     DATAWT,R2
     010314  062702  000002                                                     ADD     #2,R2
3168 010320  004773  010326                     JSR     PC,@PATTBL(R3)  ;GO GENERATE THE APPROPRIATE PATTERN.
3169 010324  000207                             RTS     PC              ;RETURN TO SETUP SUBROUTINE.
3170
3171                                            ;TK25 WRITE PATTERN LOOKUP TABLE.  USED TO JSR TO THE
3172                                            ;CORRECT DATA PATTERN GENERATING ROUTINE.
3173
3174 010326  010350                     PATTBL: PATR0                   ; INCREMENTING PATTERN, 0 - 377
3175 010330  010406                             PATR1                   ; ALL ONES PATTERN
3176 010332  010426                             PATR2                   ; ALL ZEROES PATTERN
3177 010334  010436                             PATR3                   ; '1' BIT SHIFT, RIGHT TO LEFT
3178 010336  010462                             PATR4                   ; '0' BIT SHIFT, RIGHT TO LEFT
3179 010340  010474                             PATR5                   ; ALTERNATE '0' & '1' WITH ALT. BYTES COMPL.
3180 010342  010506                             PATR6                   ; ALTERNATE BYTES OF 000 AND 377
3181 010344  010526                             PATR7                   ; RANDOM PATTERN.
3182 010346  010652                             PATR8                   ; DUMMY.  NO PATTERN, JUST EXITS.
3183
3184
3185                                            ;INCREMENTING PATTERN.  0 - 377.
3186
3187 010350                            PATR0:: LET R3 := #400
     010350  012703  000400                                                     MOV     #400,R3
3188 010354                             1$:     LET R4 := R4 - #2       ;DECREMENT WORD COUNT.
     010354  162704  000002                                                     SUB     #2,R4
3189 010360  100411                             BMI     2$              ;BR IF DONE.
3190 010362                                     LET (R2)+ := R3         ;STORE DATA WORD.
     010362  010322                                                             MOV     R3,(R2)+
3191 010364                                     LET R3 := R3 + #1002    ;UPDATE PATTERN.
     010364  062703  001002                                                     ADD     #1002,R3
3192 010370                                     IF R3 EQ #1000 THEN     ;IF PATTERN HAS WRAPPED AROUND THEN:
     010370  020327  001000                                                     CMP     R3,#1000
     010374  001002                                                             BNE     50070$
3193 010376                                       LET R3 := #400          ;INIT THE PATTERN AGAIN.
     010376  012703  000400                                                     MOV     #400,R3
3194 010402                                     ENDIF
     010402                                                                     50070$:
3195 010402  000764                             BR      1$              ;DO IT AGAIN.
3196 010404  000207                     2$:     RTS     PC              ;RETURN.
```

```
3197
3198                                        ;ALL ONE'S PATTERN.
3199
3200 010406  012703  177777        PATR1:: MOV     #-1,R3          ;ALL ONES PATTERN;.
3201 010412                        ZROPAT: LET R4 := R4 - #2       ;DECREMENT BYTE COUNT.
     010412  162704  000002                                                SUB     #2.R4
3202 010416  100402                        BMI     1$              ;DONE?.BR IF YES.
3203 010420  010322                        MOV     R3,(R2)+        ;IF NOT LOAD NEXT BYTE WITH PATTERN.
3204 010422  000773                        BR      ZROPAT          ;DO IT AGAIN.
3205
3206 010424  000207                1$:     RTS     PC              ;RETURN.
```

```
3208                                        ;ALL ZEROES PATTERN.
3209
3210 010426  005003              PATR2:: CLR     R3              ;CLR PATTERN REGISTER.
3211 010430  004737  010412              JSR     PC,ZROPAT       ;GO GENERATE IT.
3212 010434  000207                      RTS     PC              ;RETURN.
3213
3214                                        ;ONE BIT WALKING FROM R TO L IN A FIELD OF ZEROES.
3215
3216 010436  012703  000401        PATR3:: MOV     #401,R3         ;INIT PATTERN REGISTER.
3217 010442                        WLKZRO: LET R4 := R4 - #2       ;DECREMENT WORD COUNT.
     010442  162704  000002                                                  SUB     #2,R4
3218 010446  100404                      BMI     1$              ;BR IF DONE.
3219 010450  010322                      MOV     R3,(R2)+        ;LOAD DATA.
3220 010452  006303                      ASL     R3              ;SHIFT PATTERN.
3221 010454  005503                      ADC     R3              ;ADD CARRY BACK INTO PATTERN.
3222 010456  000771                      BR      WLKZRO          ;DO IT AGAIN.
3223 010460  000207              1$:     RTS     PC              ;RETURN.
3224
3225                                        ;ZERO BIT WALKING FROM R TO L IN A FIELD OF 1'S.
3226
3227 010462  012703  177376        PATR4:: MOV     #177376,R3      ;INIT PATTERN REGISTER.
3228 010466  004737  010442              JSR     PC,WLKZRO       ;GO GENERATE ;IT.
3229 010472  000207                      RTS     PC              ;RETURN.
3230
3231                                        ;ALTERNATING ONE AND ZERO BITS WITH ALTERNATE BYTES
3232                                        ;COMPLEMENTED.
3233
3234 010474  012703  125125        PATR5:: MOV     #125125,R3      ;INIT PATTERN REGISTER.
3235 010500  004737  010412              JSR     PC,ZROPAT       ;GO GENERATE IT.
3236 010504  000207                      RTS     PC              ;RETURN.
3237
3238                                        ;ALTERNATING BYTES OF 000 AND 377.
3239
3240 010506  012703  177400        PATR6:: MOV     #177400,R3      ;INIT PATTERN REGISTER.
3241 010512                        1$:     LET R4 := R4 - #2       ;DECREMENT WORD COUNT.
     010512  162704  000002                                                  SUB     #2,R4
3242 010516  100402                      BMI     2$              ;BR IF DONE.
3243 010520  010322                      MOV     R3,(R2)+        ;LOAD DATA.
3244 010522  000773                      BR      1$              ;DO IT AGAIN.
3245 010524  000207              2$:     RTS     PC              ;RETURN.
3246
3247                                        ;RANDOM PATTERN GENERATOR
3248
3249 010526                        PATR7::
3250 010526  012737  001233  010644        MOV     #RS1,RAN1       ;SET UP THE SEED
3251 010534  012737  007622  010646        MOV     #RS2,RAN2       ;SET UP THE SEED
3252 010542  012737  000000  010650        MOV     #RS3,RAN3       ;SET UP THE SEED
3253 010550                        1$:     LET R4 := R4 - #2       ;DECREMENT WORD COUNT
     010550  162704  000002                                                  SUB     #2,R4
3254 010554  100432                      BMI     GIT             ;BR IF DONE.
3255 010556  010346                      MOV     R3,-(SP)
3256 010560  013703  010644              MOV     RAN1,R3         ;MOVE THE FIRST SEED INTO R3
3257 010564  000241                      CLC                     ;CLEAR THE CARRY FLAG
3258 010566  005337  010650              DEC     RAN3            ;DECREMENT THE THIRD SEED
3259 010572  006103                      ROL     R3              ;
3260 010574  006103                      ROL     R3              ;
3261 010576  063703  010646              ADD     RAN2,R3         ;ADD THE SECOND SEED TO R3
```

```
3262 010602  010337  010644                MOV     R3,RAN1         ;PUT IT ALL IN THE FIRST SEED
3263 010606  063703  010650                ADD     RAN3,R3         ;PUT THE THIRD SEED INTO R3
3264 010612  006103                        ROL     R3              ;
3265 010614  006103                        ROL     R3              ;
3266 010616  063703  010646                ADD     RAN2,R3         ;ADD THE SECOND SEED TO R3
3267 010622  006103                        ROL     R3              ;
3268 010624  006103                        ROL     R3              ;
3269 010626  010337  010646                MOV     R3,RAN2         ;PUT IT IN THE SECONG SEED
3270 010632  012603                        MOV     (SP)+,R3        ;RESTORE R3
3271 010634  013722  010646                MOV     RAN2,(R2)+      ;PUT # IN BUFFER
3272 010640  000743                        BR      1$              ;CONTINUE
3273 010642  000207              GIT:      RTS     PC              ;RETURN
3274
3275 010644  000000              RAN1::    .WORD   0
3276 010646  000000              RAN2::    .WORD   0
3277 010650  000000              RAN3::    .WORD   0
3278         001233              RS1       **      1233
3279         007622              RS2       **      7622
3280         000000              RS3       **      0
3281
3282                             ;         NO PATTERN GENERATION.
3283
3284 010652  000207              PATR8::   RTS     PC              ;RETURN.
```

```
3286                                    ;       THIS SUBROUTINE INITIATES TK25 COMMAND EXECUTION
3287                                    ;       AND CHECKS FOR TK25 RESPONSE.
3288                                    ;       INPUTS:
3289                                    ;       OUTPUTS:
3290                                    ;       REGISTERS:      R2, R3.
3291                                    ;       CALLS:          DROPU, MOVMSG, FIRSTU, NEXTU, WSSR.
3292
3293 010654                            EXCUTE::LET TIME1 := #-1                ;INIT TIMEOUT COUNTER.
     010654  012737  177777  003444                                                     MOV     #-1,TIME1
3294 010662                                    REPEAT                          ;WAIT -
     010662                                                                             50071$:
3295 010662                                      LET TIME1 := TIME1 - #1       ;UPDATE TIMEOUT COUNTER.
     010662  005337  003444                                                            DEC     TIME1
3296 010666                                      IF TIME1 EQ #0 THEN           ;IF TIMED OUT:
     010666  005737  003444                                                            TST     TIME1
     010672  001011                                                                    BNE     50072$
3297 010674  004737  011726                        JSR PC,MOVMSG               ;MOVE CURRENT PACKET MSG.
3298 010700                                        ERRDF 2,NSSRM,STAERM        ;REPORT TK25 NOT READY
     010700  104455                                                                    TRAP    C$ERDF
     010702  000002                                                                    .WORD   2
     010704  004353                                                                    .WORD   NSSRM
     010706  005626                                                                    .WORD   STAERM
3299 010710  004737  017402                        JSR PC,DROPU                ;DROP THE UNIT.
3300 010714  000566                                BR  EXCRTN                  ;RETURN.
3301 010716                                      ENDIF
     010716                                                                             50072$:
3302 010716                                    UNTIL #TS.SSR SETIN @TSSR(R5)   ;WAIT UNTIL DEVICE IS READY.
     010716  032775  000200  002466                                                    BIT     #TS.SSR,@TSSR(R5)
     010724  001756                                                                    BEQ     50071$
3303 010726                                    IF CMDWRD EQ #SCH THEN          ;IF WE ARE DOING A SET CHAR CMD THEN:
     010726  023727  003426  140004                                                    CMP     CMDWRD,#SCH
     010734  001022                                                                    BNE     50073$
3304 010736                                      LET R5SAVE := R5              ;SAVE CURRENT DEVICE POINTER.
     010736  010537  003460                                                            MOV     R5,R5SAVE
3305 010742  004737  017300                      JSR   PC,FIRSTU               ;FIND FIRST UNIT.
3306 010746                                      WHILE DEVTBL(R5) NE #END DO
     010746                                                                             50074$:
     010746  026527  002536  177777                                                    CMP     DEVTBL(R5),#END
     010754  001405                                                                    BEQ     50075$
3307 010756  004737  011672                        JSR PC,WSSR                 ;WAIT FOR UNIT READY OR TIME OUT,
3308 010762  004737  017346                        JSR PC,NEXTU                ;FIND NEXT UNIT.
3309 010766                                      ENDDO
     010766  000767                                                                    BR      50074$
     010770                                                                             50075$:
3310 010770                                    LET R5 := R5SAVE                ;RESTORE CURRENT DEVICE POINTER.
     010770  013705  003460                                                            MOV     R5SAVE,R5
3311 010774                                    LET SCHBK := MSGPKA(R5)         ;SET UP ADR OF MSG PKT IN SCH BLOCK.
     010774  016537  002506  002446                                                    MOV     MSGPKA(R5),SCHBK
3312 011002                                    ENDIF
     011002                                                                             50073$:
3313 011002                                    LET R3 := MSGPKA(R5)            ;ADR OF THIS UNIT'S MSG PACKET.
     011002  016503  002506                                                            MOV     MSGPKA(R5),R3
3314 011006                                    LET R2 := #0                    ;CLR COUNTER.
     011006  005002                                                                    CLR     R2
3315 011010                                    WHILE R2 NE #MSGCNT DO          ;WHILE THERE ARE MORE LOCATIONS:
     011010                                                                             50076$:
     011010  020227  000016                                                            CMP     R2,#MSGCNT
```

```
     011014  001405                                                                        BEQ     50077$
3316 011016                          LET (R3)+ := @-1                 ;INIT THE MSG PACKET WITH ALL 1'S
     011016  012723  177777                                                                MOV     @-1,(R3)+
3317 011022                          LET R2 := R2 + @2                ;UPDATE COUNTER.
     011022  062702  000002                                                                ADD     @2,R2
3318 011026                        ENDDO
     011026  000770                                                                        BR      50076$
     011030                                                                                50077$:
3319 011030  105737  002212        TSTB    DINT                       ;ARE INTERRUPTS DISABLED.
3320 011034  001023                BNE     1$                         ;BR IF YES.
3321 011036                        IFB INTFLG(R5) GT @1 THEN          ;IF MORE THAN ONE INTERRUPT HAS OCCURED:
     011036  126527  003476  000001                                                        CMPB    INTFLG(R5),@1
     011044  003412                                                                        BLE     50100$
3322 011046                          LET TSSREG := @TSSR(R5)                    ;FREEZE THE CURRENT STATUS REG FOR PRINT
     011046  017537  002466  003462                                                        MOV     @TSSR(R5),TSSREG
3323 011054                          ERRDF 15,TOOMM,STAERM   ;REPORT TOO MANY INTERRUPTS.
     011054  104455                                                                                TRAP    C$ERDF
     011056  000017                                                                                .WORD   15
     011060  004542                                                                                .WORD   TOOMM
     011062  005626                                                                                .WORD   STAERM
3324 011064  004737  017402          JSR PC,DROPU                      ;DROP THE UNIT
3325 011070  000500                  BR  EXCRTN                        ;RETURN - UNIT HAS BEEN DROPPED.
3326 011072                        ENDIF
     011072                                                                                50100$:
3327 011072                        LET INTFLG(R5) := @0               ;CLR INTERRUPT FLAG FOR THIS DEV.
     011072  005065  003476                                                                CLR     INTFLG(R5)
3328 011076  052737  000200  002314        BIS     @IE.C,CMDPKT                ;SET INT ENABLE BIT.
3329 011104                    1$:  IFB ERRREC EQ @0 THEN             ;IF NOT RETRYING
     011104  105737  003475                                                                TSTB    ERRREC
     011110  001005                                                                        BNE     50101$
3330 011112                          LET RECCNT(R5) := RECCNT(R5) + @1
     011112  005265  003330                                                                INC     RECCNT(R5)
3331 011116                          LET @DATAWT := RECCNT(R5)       ;THEN UPDATE REC COUNT TO WRITE IT ON TAPE
     011116  016577  003330  172270                                                        MOV     RECCNT(R5),@DATAWT
3332 011124                        ENDIF
     011124                                                                                50101$:
3333 011124                        IF L$TEST EQ @3 AND CMDWRD EQ @WRT THEN
     011124  023727  002114  000003                                                        CMP     L$TEST,@3
     011132  001040                                                                        BNE     50102$
     011134  023727  003426  104005                                                        CMP     CMDWRD,@WRT
     011142  001034                                                                        BNE     50102$
3334 011144                        LET LOOPCT :B= LOOPCT + 1
     011144  005046                                                                        CLR     -(SP)
     011146  113716  003525                                                                MOVB    LOOPCT,(SP)
     011152  063716  000001                                                                ADD     1,(SP)
     011156  111637  003525                                                                MOVB    (SP),LOOPCT
     011162  062706  000002                                                                ADD     @2,SP
3335 011166                         IFB LOOPCT GT 3 THEN
     011166  123737  003525  000003                                                        CMPB    LOOPCT,3
     011174  003417                                                                        BLE     50103$
3336 011176                          DELAY 21.
     011176  012727  000025                                                                        MOV     @21.,(PC)+
     011202  000000                                                                                .WORD   0
     011204  013727  002116                                                                        MOV     L$DLY,(PC)+
     011210  000000                                                                                .WORD   0
     011212  005367  177772                                                                        DEC     -6(PC)
     011216  001375                                                                                BNE     .-4
```

```
                011220  005367  177756                                                          DEC     -22(PC)
                011224  001367                                                                  BNE     .-20
          3337  011226                          LET LOOPCT ;B= 0
                011226  113737  000000  003525                                                  MOVB    0,LOOPCT
          3338  011234                          ENDIF
                011234                                                                          50103$:
          3339  011234                        ENDIF
                011234                                                                          50102$:
          3340  011234  012775  002314  002456        MOV     @CMDPKT,@TSDB(R5)       ;LOAD TSDB WITH CMDPKT ADDRESS
          3341                                                                        ;THIS INITIATES COMMAND EXECUTION.
          3342  011242                        IF @TS.SSR SETIN @TSSR(R5) THEN ;IF READY DID NOT DROP THEN:
                011242  032775  000200  002466                                                  BIT     @TS.SSR,@TSSR(R5)
                011250  001410                                                                  BEQ     50104$
          3343  011252  004737  011726          JSR   PC,MOVMSG                 ;MOVE CURRENT MESSAGE PACKET TO COMMON.
          3344  011256          ERRDF 3,TOERM,STAERM            ;REPORT NO TK25 RESPONSE.
                011256  104455                                                                  TRAP    C$ERDF
                011260  000003                                                                  .WORD   3
                011262  004271                                                                  .WORD   TOERM
                011264  005626                                                                  .WORD   STAERM
          3345  011266  004737  017402          JSR PC,DROPU                    ;DROP THE UNIT
          3346  011272                        ENDIF
                011272                                                                          50104$:
          3347  011272  000207                EXCRTN: RTS     PC                      ;RETURN.
```

```
                        ;       THIS SUBROUTINE WAITS FOR THE TK25 INERRUPT OR DONE BIT TO SET AND ALLOWS THE
                        ;       OPERATOR TO TRANSFER CONROL TO THE SUPERVISOR.
                        ;       UPON APPEARANCE OF THE INTERRUPT OR DONE, CHECK TSSR FOR STATUS ERRORS,
                        ;       LOG BYTES AND ERRORS AND PERFORM ERROR RECOVERY IF NESSASARY.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2, R3.
                        ;       CALLS:          DROPU, MOVMSG, RECUD, CHKERR, LOG, CLRERR.

011274                  GOWAIT::IF DEVTBL(R5) EQ #NINUSE THEN
011274  026527  002536  177774                                          CMP     DEVTBL(R5),#NINUSE
011302  001001                                                          BNE     50105$
011304  000563          BR      1$
011306                  ENDIF
011306                                                                  50105$:
011306                  LET TIME1 := #-1                ;INIT TIME OUT COUNTER.
011306  012737  177777  003444                                          MOV     #-1,TIME1
011314                  REPEAT                          ;REPEAT UNTIL INTERRUPT OCCURES:
011314                                                                  50106$:
011314                   BREAK                          ;HONOR SUPERVISOR BREAKS
011314  104422                                                          TRAP    C$BRK
011316                   IF PCMDWD EQ #RWD AND CMDWRD EQ #WRT THEN
011316  023727  003432  102010                                          CMP     PCMDWD,#RWD
011324  001020                                                          BNE     50107$
011326  023727  003426  104005                                          CMP     CMDWRD,#WRT
011334  001014                                                          BNE     50107$
                                                        ;POSSIBLE FIRST WRITE ON TAPE
011336                    DELAY 40.
011336  012727  000050                                                  MOV     #40.,(PC)+
011342  000000                                                          .WORD   0
011344  013727  002116                                                  MOV     L$DLY,(PC)+
011350  000000                                                          .WORD   0
011352  005367  177772                                                  DEC     -6(PC)
011356  001375                                                          BNE     .-4
011360  005367  177756                                                  DEC     -22(PC)
011364  001367                                                          BNE     .-20
011366                  ENDIF                           ;SO DELAY TO CALIBRATE TAPE
011366                                                                  50107$:
011366                   IF CMDWRD EQ #RWD THEN         ;IF COMMAND WAS REWIND THEN:
011366  023727  003426  102010                                          CMP     CMDWRD,#RWD
011374  001014                                                          BNE     50110$
011376                     DELAY 10.                    ;WAIT  EXTRA 10 MSECS EACH LOOP.
011376  012727  000012                                                  MOV     #10.,(PC)+
011402  000000                                                          .WORD   0
011404  013727  002116                                                  MOV     L$DLY,(PC)+
011410  000000                                                          .WORD   0
011412  005367  177772                                                  DEC     -6(PC)
011416  001375                                                          BNE     .-4
011420  005367  177756                                                  DEC     -22(PC)
011424  001367                                                          BNE     .-20
011426                   ENDIF
011426                                                                  50110$:
011426                   IF CMDWRD EQ #SFF OR CMDWRD EQ #SFR THEN
011426  023727  003426  105010                                          CMP     CMDWRD,#SFF
011434  001404                                                          BEQ     50111$
011436  023727  003426  105410                                          CMP     CMDWRD,#SFR
011444  001014                                                          BNE     50112$
```

E7


```
      011446                                                                    50111$:
 3372 011446                                 DELAY 12.                 ;ADD DELAY FOR SPACE TAPE MARK COMMANDS
      011446  012727  000014                                                            MOV     #12.,(PC)+
      011452  000000                                                                    .WORD   0
      011454  013727  002116                                                            MOV     L$DLY,(PC)+
      011460  000000                                                                    .WORD   0
      011462  005367  177772                                                            DEC     -6(PC)
      011466  001375                                                                    BNE     .-4
      011470  005367  177756                                                            DEC     -22(PC)
      011474  001367                                                                    BNE     .-20
 3373 011476                               ENDIF
      011476                                                                    50112$:
 3374 011476                               IFB DINT EQ #0 THEN           ;IF INTERRUPTS ARE ENABLED.
      011476  105737  002212                                            TSTB    DINT
      011502  001003                                                    BNE     50113$
 3375 011504                                 LET R2 := INTFLG(R5)        ;FETCH INTERRUPT OCCURRED FLAG.
      011504  016502  003476                                            MOV     INTFLG(R5),R2
 3376 011510                               ELSE                          ;IF IN BRUTUS MODE:
      011510  000406                                                    BR      50114$
      011512                                                            50113$:
 3377 011512                                 LET R3 := COMP #TS.SSR      ;SET UP A MASK FOR THE DONE BIT.
      011512  012703  000200                                            MOV     #TS.SSR,R3
      011516  005103                                                    COM     R3
 3378 011520                                 LET R2 := @TSSR(R5) CLR.BY R3  ;FETCH DONE BIT.
      011520  017502  002466                                            MOV     @TSSR(R5),R2
      011524  040302                                                    BIC     R3,R2
 3379 011526                               ENDIF
      011526                                                            50114$:
 3380 011526                               LET TIME1 := TIME1 - #1       ;UPDATE TIMEOUT COUNTER.
      011526  005337  003444                                            DEC     TIME1
 3381 011532                             UNTIL R2 NE #0 OR TIME1 EQ #0   ;REPEAT UNTIL INTERRUPT OR READY OCCURES.
      011532  005702                                                    TST     R2
      011534  001003                                                    BNE     50115$
      011536  005737  003444                                            TST     TIME1
      011542  001264                                                    BNE     50106$
      011544                                                            50115$:
 3382 011544                             IF TIME1 EQ #0 THEN             ;IF TIME OUT HAS OCCURRED:
      011544  005737  003444                                            TST     TIME1
      011550  001022                                                    BNE     50116$
 3383 011552                               LET @DATAWT := RECCNT(R5) - #1 ;RE-ADJUST REC COUNT DOWN
      011552  016577  003330  171634                                    MOV     RECCNT(R5),@DATAWT
      011560  005377  171630                                            DEC     @DATAWT
 3384 011564  004737  011726               JSR   PC,MOVMSG               ;MOVE CURRENT MSG PACKET TO COMMON AREA.
 3385 011570                               ERRDF 4,NOINTM,STAERM ;REPORT NO INTERRUPT.
      011570  104455                                                            TRAP    C$ERDF
      011572  000004                                                            .WORD   4
      011574  004503                                                            .WORD   NOINTM
      011576  005626                                                            .WORD   STAERM
 3386 011600  004737  017402               JSR PC,DROPU             ;DROP THE UNIT.
 3387 011604                               LET R3 := #ENDERF
      011604  012703  003476                                            MOV     #ENDERF,R3
 3388 011610  004737  011656               JSR PC,CLRERR                 ;CLEAR ALL ERROR FLAGS
 3389 011614                             ELSE
      011614  000417                                                    BR      50117$
      011616                                                            50116$:
 3390 011616  004737  011726               JSR   PC,MOVMSG               ;MOVE CURRENT MSG. PACKET TO COMMON AREA.
 3391 011622  004737  012012               JSR   PC,RECUD                ;UPDATE THE RECORD COUNT.
```

```
3392 011626  004737  012160                    JSR   PC,CHKERR              ;CHECK FOR STATUS ERRORS.
3393 011632                                    IFB WRTYFG EQ #0 THEN          ;
     011632  105737  003467                                                    TSTB    WRTYFG
     011636  001006                                                          BNE     50120$
3394 011640  004737  015436                      JSR PC,LOG                 ;LOG BYTES AND ERRORS.
3395 011644                                      LET R3 := #ENDERF
     011644  012703  003476                                                  MOV     #ENDERF,R3
3396 011650  004737  011656                      JSR PC,CLRERR                      ;CLEAR ALL ERROR FLAGS
3397 011654                                    ENDIF
     011654                                                                  50120$:
3398 011654                                  ENDIF
     011654                                                                  50117$:
3399 011654  000207                  1$:     RTS     PC                     ;RETURN IF DONE.
```

```
3401                                    ;       SUBROUTINE TO CLEAR FLAGS.
3402                                    ;       INPUTS:         R3 = LWA TO BE CLEARED + 2.
3403                                    ;       OUTPUTS:
3404                                    ;       REGISTERS:      R2
3405                                    ;       CALLS:
3406
3407 011656                            CLRERR:: LET R2 := #BGNFLG
     011656  012702  003464                                                         MOV     #BGNFLG,R2
3408 011662                                     REPEAT
     011662                                                                         50121$:
3409 011662                                       LET (R2)+ := #0
     011662  005022                                                                 CLR     (R2)+
3410 011664                                     UNTIL R2 EQ R3
     011664  020203                                                                 CMP     R2,R3
     011666  001375                                                                 BNE     50121$
3411 011670  000207                            RTS PC
3412
3413
3414
3415                                    ;       SUBROUTINE TO WAIT UNTIL CURRENT UNIT IS READY OR UNTIL TIME OUT.
3416                                    ;       INPUTS:
3417                                    ;       OUTPUTS:
3418                                    ;       REGISTERS:
3419                                    ;       CALLS:
3420
3421 011672                            WSSR::  LET TIME1 := #-1                ;INIT TIMEOUT COUNTER.
     011672  012737  177777  003444                                                 MOV     #-1,TIME1
3422 011700                                     REPEAT                          ;REPEAT UNTIL DEV READY OR TIMEOUT:
     011700                                                                         50122$:
3423 011700                                       BREAK                         ;BREAK TO THE SUPERVISOR.
     011700  104422                                                                 TRAP    C$BRK
3424 011702                                       LET TIME1 := TIME1 - #1       ;UPDATE TIMEOUT COUNTER.
     011702  005337  003444                                                         DEC     TIME1
3425 011706                                     UNTIL #TS.SSR SETIN @TSSR(R5) OR TIME1 EQ #0
     011706  032775  000200  002466                                                 BIT     #TS.SSR,@TSSR(R5)
     011714  001003                                                                 BNE     50123$
     011716  005737  003444                                                         TST     TIME1
     011722  001366                                                                 BNE     50122$
     011724                                                                         50123$:
3426                                                                    ;REPEAT UNTIL DEV READY OR TIMEOUT.
3427 011724  000207                            RTS PC                  ;RETURN.
3428
3429
```

```
3431
3432                                    ;       SUBROUTINE TO MOVE THE CURRENT MESSAGE PACKET TO THE COMMON AREA AND
3433                                    ;       TO UPDATE THE CURRENT TERMINATION CLASS CODE.
3434                                    ;       INPUTS:
3435                                    ;       OUTPUTS:
3436                                    ;       REGISTERS:      R2, R3.
3437                                    ;       CALLS:
3438
3439 011726                            MOVMSG:: LET TSSREG := @TSSR(R5)          ;FREEZE THE STATUS REG CONTENTS
     011726  017537  002466  003462                                                MOV     @TSSR(R5),TSSREG
3440 011734                                     LET R2 := TSSREG CLR.BY #TSC.TCC ;EXTRACT THE TERMINATION CLASS CODE.
     011734  013702  003462                                                       MOV     TSSREG,R2
     011740  042702  177761                                                       BIC     #TSC.TCC,R2
3441 011744                                     LET CTCC := R2 SHIFT -1          ;AND SAVE IT
     011744  010237  003456                                                       MOV     R2,CTCC
     011750  006237  003456                                                       ASR     CTCC
3442 011754                                     LET R3 := MSGPKA(R5)             ;ADR OF THIS DEVICE'S MSG.
     011754  016503  002506                                                       MOV     MSGPKA(R5),R3
3443 011760                                     LET R2 := #0                     ;CLR COUNTER.
     011760  005002                                                               CLR     R2
3444 011762                                     WHILE R2 NE #MSGCNT DO           ;WHILE THERE ARE MORE LOCATIONS:
     011762                                                                     50124$:
     011762  020227  000016                                                       CMP     R2,#MSGCNT
     011766  001405                                                               BEQ     50125$
3445 011770                                       LET MSGPKT(R2) := (R3)+        ;MOVE MSG TO COMMON AREA.
     011770  012362  002340                                                       MOV     (R3)+,MSGPKT(R2)
3446 011774                                       LET R2 := R2 + #2              ;UPDATE COUNTER.
     011774  062702  000002                                                       ADD     #2,R2
3447 012000                                     ENDDO
     012000  000770                                                               BR      50124$
     012002                                                                     50125$:
3448 012002                                     LET EOTFLG(R5) := MSGPKT+MS.XSO ;MOVE XSTATO TO EOT FLAG.
     012002  013765  002346  003506                                               MOV     MSGPKT+MS.XSO,EOTFLG(R5)
3449 012010  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:
012012  105737  003471                                                        TSTB    RECLOG
012016  001057                                                              BNE     50126$
                                  LET RECCNT(R5) := RECCNT(R5) - #1
012020  005365  003330                                                      DEC     RECCNT(R5)
                                  IF #BIT0 NOTSETIN CTCC AND #X2.OPM SETIN MSGPKT+MS.XS2 THEN   ;IF TAPE MOVED THEN:
012024  032737  000001  003456                                              BIT     #BIT0,CTCC
012032  001046                                                              BNE     50127$
012034  032737  100000  002352                                              BIT     #X2.OPM,MSGPKT+MS.XS2
012042  001442                                                              BEQ     50127$
                                    LET RECLOG :B= RECLOG + #1  ;SET RECORD LOGGED,
012044  105237  003471                                                      INCB    RECLOG
                                    IF CMDWRD EQ #RWD THEN      ;IF THIS IS A REWIND CMD:
012050  023727  003426  102010                                              CMP     CMDWRD,#RWD
012056  001003                                                              BNE     50130$
                                      LET RECCNT(R5) := #0      ;CLEAR RECORD COUNT,
012060  005065  003330                                                      CLR     RECCNT(R5)
                                    ELSE
012064  000431                                                              BR      50131$
012066                                                                      50130$:
                                      IF #BRF.C SETIN CMDWRD THEN       ;IF BRF USED, UPDATE RECORD COUNT.
012066  032737  004000  003426                                              BIT     #BRF.C,CMDWRD
012074  001425                                                              BEQ     50132$
                                      IF #MOD.CO NOTSETIN CMDWRD THEN   ;IF A FORWARD CMD:
012076  032737  000400  003426                                              BIT     #MOD.CO,CMDWRD
012104  001007                                                              BNE     50133$
                                        IF #MOD.CO NOTSETIN PCMDWD THEN ;IF PREV CMD WAS A FWD ALSO:
012106  032737  000400  003432                                              BIT     #MOD.CO,PCMDWD
012114  001002                                                              BNE     50134$
                                          LET RECCNT(R5) := RECCNT(R5) + #1  ;INCREMENT RECORD COUNT.
012116  005265  003330                                                      INC     RECCNT(R5)
                                        ENDIF
012122                                                                      50134$:
                                      ELSE                          ;IF REVERSE CMD:
012122  000412                                                              BR      50135$
012124                                                                      50133$:
                                        IF #MOD.CO SETIN PCMDWD THEN  ;IF PREVIOUS CMD WAS A REV ALSO:
012124  032737  000400  003432                                              BIT     #MOD.CO,PCMDWD
012132  001406                                                              BEQ     50136$
                                          IF #X0.BOT NOTSETIN EOTFLG(R5) THEN ;WHEN NOT AT BOT THEN
012134  032765  000002  003506                                              BIT     #X0.BOT,EOTFLG(R5)
012142  001002                                                              BNE     50137$
                                            LET RECCNT(R5) := RECCNT(R5) - #1  ;DECREMENT RECORD COUNT.
012144  005365  003330                                                      DEC     RECCNT(R5)
                                        ENDIF
012150                                                                      50137$:
                                      ENDIF
012150                                                                      50136$:
                                    ENDIF
012150                                                                      50135$:
                                ENDIF
```

```
     012150                                                                  50132$:
3477 012150                                ENDIF
     012150                                                                  50131$:
3478 012150                              ENDIF
     012150                                                                  50127$:
3479 012150                              LET &DATAWT := RECCNT(R5)
     012150  016577  003330  171236                                         MOV     RECCNT(R5),&DATAWT
3480 012156                            ENDIF
     012156                                                                  50126$:
3481 012156  000207                       RTS     PC                  ;RETURN.
```

```
                                ;       THIS IS THE ERROR CHECK SUBROUTINE.  AFTER INTERRUPT THIS
                                ;       SUBROUTINE IS CALLED TO CHECK THE TK25 STATUS.
                                ;       IF SPECIAL COND IS SET THEN THE TCC HANDLING SUBROUTINE IS ENTERED.
                                ;       IF THE RFC IS NON ZERO FOR A COMMAND REQUIRING A BPCR,
                                ;       THEN AN ERROR RFC IS REPORTED.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R4.
                                ;       CALLS:          TCC0-TCC7.

3493 012160                     CHKERR::IF DEVTBL(R5) NE #NINUSE THEN
     012160  026527  002536  177774                                          CMP     DEVTBL(R5),#NINUSE
     012166  001556                                                         BEQ     50140$
3494 012170                             IF #X1.EWN SETIN MSGPKT+MS.XS1 THEN     ;IF EARLY WARNING IS SET AND
     012170  032737  000010  002350                                         BIT     #X1.EWN,MSGPKT+MS.XS1
     012176  001447                                                         BEQ     50141$
3495 012200                               IFB EWPRNT NE #0 THEN         ;IF END OF TRACK STATUS PRINTS ALLOWED
     012200  105737  002216                                                TSTB    EWPRNT
     012204  001421                                                         BEQ     50142$
3496 012206                               LET R2 := MSGPKT+MS.XS1 CLR.BY #177417
     012206  013702  002350                                                MOV     MSGPKT+MS.XS1,R2
     012212  042702  177417                                                BIC     #177417,R2
3497 012216                               LET R2 := R2 SHIFT -4
     012216  006202                                                         ASR     R2
     012220  006202                                                         ASR     R2
     012222  006202                                                         ASR     R2
     012224  006202                                                         ASR     R2
3498 012226                               PRINTF #EWMSG,R2              ;PRINT END OF TRACK MESSAGE
     012226  010246                                                         MOV     R2,-(SP)
     012230  012746  012526                                                MOV     #EWMSG,-(SP)
     012234  012746  000002                                                MOV     #2,-(SP)
     012240  010600                                                         MOV     SP,R0
     012242  104417                                                         TRAP    C$PNTF
     012244  062706  000006                                                ADD     #6,SP
3499 012250                             ENDIF
     012250                                                             50142$:
3500 012250                             IF CTCC EQ #4 THEN              ;IF TCC4 AND WE DID A WRITE
     012250  023727  003456  000004                                         CMP     CTCC,#4
     012256  001017                                                         BNE     50143$
3501 012260                               IF CMDWRD EQ #WRT OR CMDWRD EQ #WTM THEN
     012260  023727  003426  104005                                         CMP     CMDWRD,#WRT
     012266  001404                                                         BEQ     50144$
     012270  023727  003426  100011                                         CMP     CMDWRD,#WTM
     012276  001007                                                         BNE     50145$
     012300                                                             50144$:
3502 012300                                 IFB IREC EQ #0 THEN
     012300  105737  002210                                                TSTB    IREC
     012304  001004                                                         BNE     50146$
3503 012306  004737  012632                   JSR     PC,EWRTRY      ;DO EW RETRY
3504 012312  000137  012524                   JMP     1$             ;ALAS, A "GOTO" STATEMENT, EH?
3505 012316                                 ENDIF
     012316                                                             50146$:
3506 012316                               ENDIF
     012316                                                             50145$:
3507 012316                             ENDIF
     012316                                                             50143$:
3508 012316                     ENDIF
```

```
     012316
3509 012316                    IF @TS.SC SETIN TSSREG THEN      ;IF SPECIAL COND STATUS IS SET THEN:
     012316  032737  100000  003462                                    BIT     @TS.SC,TSSREG
     012324  001441                                                    BEQ     50147$
3510 012326                      IF CTCC NE #2 THEN             ;IF TCC IS NOT 2 THEN:
     012326  023727  003456  000002                                    CMP     CTCC,#2
     012334  001405                                                    BEQ     50150$
3511 012336                        IFB ERRREC EQ #0 THEN        ;IF NOT IN ERROR RECOVERY:
     012336  105737  003475                                          TSTB    ERRREC
     012342  001002                                                    BNE     50151$
3512 012344  005265  003270          INC     SCCNT(R5)        ;INC SC COUNTER.
3513 012350                        ENDIF
     012350                                                    50151$:
3514 012350                      ENDIF
     012350                                                    50150$:
3515 012350                      IF @TS.NXM SETIN TSSREG OR @TS.UPE SETIN TSSREG THEN  ;WHEN NON-EXISTANT MEMO
     012350  032737  004000  003462                                    BIT     @TS.NXM,TSSREG
     012356  001004                                                    BNE     50152$
     012360  032737  040000  003462                                    BIT     @TS.UPE,TSSREG
     012366  001412                                                    BEQ     50153$
     012370                                                    50152$:
3516 012370                        IF @X2.OPM NOTSETIN MSGPKT+MS.XS2 THEN          ;AND TAPE NOT MOVED
     012370  032737  100000  002352                                    BIT     @X2.OPM,MSGPKT+MS.XS2
     012376  001003                                                    BNE     50154$
3517 012400                          LET R2 := #5               ;SET TCC5 INDEX
     012400  012702  000005                                          MOV     #5,R2
3518 012404                        ELSE
     012404  000402                                                    BR      50155$
     012406                                                    50154$:
3519 012406                          LET R2 := #4               ;TAPE MOVED, SET TCC4 INDEX
     012406  012702  000004                                          MOV     #4,R2
3520 012412                        ENDIF
     012412                                                    50155$:
3521 012412                      ELSE
     012412  000402                                                    BR      50156$
     012414                                                    50153$:
3522 012414                        LET R2 := CTCC               ;SET DETECTED TCC INDEX
     012414  013702  003456                                          MOV     CTCC,R2
3523 012420                      ENDIF
     012420                                                    50156$:
3524 012420                      LET R2 := R2 SHIFT 1   ;CURRENT TCC X 2.
     012420  006302                                                  ASL     R2
3525 012422  004772  012612      JSR   PC,@TCCRA(R2)              ;GO TO THE TCC HANDLING SUBROUTINE.
3526 012426                    ELSE
     012426  000430                                                  BR      50157$
     012430                                                    50147$:
3527 012430                      IF @BRF.C SETIN CMDWRD THEN    ;IF BRF IS USED IN THIS CMD THEN:
     012430  032737  004000  003426                                    BIT     @BRF.C,CMDWRD
     012436  001424                                                    BEQ     50160$
3528 012440                        IF MSGPKT+MS.RFC NE #0 THEN    ;IF THERE IS AN RFC THEN:
     012440  005737  002344                                          TST     MSGPKT+MS.RFC
     012444  001421                                                    BEQ     50161$
3529 012446                          IFB RANDOM EQ #0 ORB VFYFLG NE #0 THEN
     012446  105737  003521                                          TSTB    RANDOM
     012452  001403                                                    BEQ     50162$
     012454  105737  003522                                          TSTB    VFYFLG
     012460  001413                                                    BEQ     50163$
```

```
                                                                                50162$:
3530                                                                    ;IF NOT IN RANDOM OR IF CMD IS WTV:
3531 012462                     IFB IRE EQ #0 THEN                      ;IF RFC ERROR REPORTS ARE ALLOWED:
     012462  105737  003526                                                      TSTB    IRE
     012466  001010                                                            BNE     50164$
3532 012470                     LET HRDCNT(R5) := HRDCNT(R5) + #1   ;UPDATE HARD ERROR COUNT
     012470  005265  003310                                                     INC     HRDCNT(R5)
3533 012474                     ERRHRD 13,RFCERM,STAERM ;REPORT RFC ERROR
     012474  104436                                                            TRAP    C$ERHRD
     012476  000015                                                            .WORD   13
     012500  004336                                                            .WORD   RFCERM
     012502  005626                                                            .WORD   STAERM
3534 012504  004737  014050             JSR     PC,RAMDUM               ;GO DO RAM DUMP
3535 012510                     ENDIF
     012510                                                                 50164$:
3536 012510                   ENDIF
     012510                                                                 50163$:
3537 012510                 ENDIF
     012510                                                                 50161$:
3538 012510               ENDIF
     012510                                                                 50160$:
3539 012510             ENDIF
     012510                                                                 50157$:
3540 012510             IFB RWERR NE #0 THEN            ;IF A READ/WRITE ERROR HAS OCCURRED THEN:
     012510  105737  003473                                                      TSTB    RWERR
     012514  001403                                                            BEQ     50165$
3541 012516               LET CMDPKT := CMDSAV            ;RESTORE CMD PACKET AFTER ERROR RECOV.
     012516  013737  003430  002314                                             MOV     CMDSAV,CMDPKT
3542 012524             ENDIF
     012524                                                                 50165$:
3543 012524                     ENDIF
     012524                                                                 50140$:
3544 012524  000207             1$:     RTS     PC                      ;RETURN.
3545
3546 012526     045     116     045  EWMSG:  .ASCIZ  /#N#AEARLY WARNING DURING WRITE AT END OF TRACK #D2/
     012531     101     105     101
     012534     122     114     131
     012537     040     127     101
     012542     122     116     111
     012545     116     107     040
     012550     104     125     122
     012553     111     116     107
     012556     040     127     122
     012561     111     124     105
     012564     040     101     124
     012567     040     105     116
     012572     104     040     117
     012575     106     040     124
     012600     122     101     103
     012603     113     040     045
     012606     104     062     000
3547                            .EVEN
```

```
3549                                    ;       ADDRESSES OF TCC HANDLING ROUTINES FOR TERMINATION CLASS CODES 0 - 7.
3550
3551 012612  012740                     TCCRA:  TCC0
3552 012614  012762                             TCC1
3553 012616  013000                             TCC2
3554 012620  013146                             TCC3
3555 012622  013170                             TCC4
3556 012624  013670                             TCC5
3557 012626  013772                             TCC6
3558 012630  014026                             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).
                        ;
                        EWRTRY::
                                LET -(SP) := CMDPKT             ;SAVE THE COMMAND PACKET
012632  013746  002314                                                MOV     CMDPKT,-(SP)
                                LET CMDPKT := CMDPKT SET.BY #MOD.C1     ;SET THE RETRY MODIFIER IN COMMAND
012636  052737  001000  002314                                        BIS     #MOD.C1,CMDPKT
                                LET ERRREC :B= #1               ;SHOW ERROR RECOVERY IN PROCESS
012644  112737  000001  003475                                        MOVB    #1,ERRREC
                                JSR     PC,EXCUTE               ;DO THE COMMAND
012652  004737  010654
                                LET ERRREC :B= #0               ;NO MORE ERROR RECOVERY
012656  105037  003475                                                CLRB    ERRREC
                                LET CMDPKT := (SP)+             ;RESTORE THE ORIGINAL COMMAND STATUS
012662  012637  002314                                                MOV     (SP)+,CMDPKT
                                LET R2 := (SP)+                 ;MODIFY THE RETURN ADDRESS
012666  012602                                                        MOV     (SP)+,R2
                                LET -(SP) := #10.               ;OVERALL LOOP CONTROL
012670  012746  000012                                                MOV     #10.,-(SP)
                                REPEAT                          ;LOOP
012674                                                          50166$:
                                DELAY 250.                      ;WAIT FOR THE UNIT TO STOP
012674  012727  000372                                                MOV     #250.,(PC)+
012700  000000                                                        .WORD   0
012702  013727  002116                                                MOV     L$DLY,(PC)+
012706  000000                                                        .WORD   0
012710  005367  177772                                                DEC     -6(PC)
012714  001375                                                        BNE     .-4
012716  005367  177756                                                DEC     -22(PC)
012722  001367                                                        BNE     .-20
                                LET (SP) := (SP) - #1           ;ONE LESS ITERATION
012724  005316                                                        DEC     (SP)
                                UNTIL (SP) EQ #0                ;UNTIL = 0
012726  005716                                                        TST     (SP)
012730  001361                                                        BNE     50166$
                                LET SP := SP + #2               ;CORRECT THE STACK
012732  062706  000002                                                ADD     #2,SP
                                RTS     PC                      ;RETURN TO CALLER
012736  000207
```

```
3583                                    ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 0, UNDEFINED SPECIAL
3584                                    ;       CONDITION ERROR.
3585                                    ;       INPUTS:
3586                                    ;       OUTPUTS:
3587                                    ;       REGISTERS:
3588                                    ;       CALLS:
3589
3590 012740                            TCC0::  LET HRDCNT(R5) := HRDCNT(R5) + #1  ;UPDATE HARD ERROR COUNT.
     012740  005265  003310                                                        INC     HRDCNT(R5)
3591 012744                                    ERRHRD 5,SCERM,STAERM               ;REPORT SPECIAL CONDITION ERROR.
     012744  104456                                                                TRAP    C$ERHRD
     012746  000005                                                                .WORD   5
     012750  004312                                                                .WORD   SCERM
     012752  005626                                                                .WORD   STAERM
3592 012754  004737  014050                    JSR     PC,RAMDUM                   ;GO DO RAM DUMP
3593 012760  000207                            RTS PC                              ;RETURN.
3594
3595
3596
3597
3598
3599                                    ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 1, ATTENTION CONDITION.
3600                                    ;       THIS TCC INDICATES THAT THE DRIVE HAS UNDERGONE A STATUS CHANGE
3601                                    ;       SUCH AS GOING OFFLINE OR COMING ONLINE.
3602                                    ;       INPUTS:
3603                                    ;       OUTPUTS:
3604                                    ;       REGISTERS:      R2,R4
3605                                    ;       CALLS:          DROPU
3606
3607 012762                            TCC1::  ERRDF 6,ATTNM,STAERM                ;REPORT ATTENTION-UNIT OFF LINE.
     012762  104455                                                                TRAP    C$ERDF
     012764  000006                                                                .WORD   6
     012766  004417                                                                .WORD   ATTNM
     012770  005626                                                                .WORD   STAERM
3608 012772  004737  014050                    JSR     PC,RAMDUM                   ;GO DO RAM DUMP
3609 012776  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:

013000                          TCC2::  IF #XO.BOT SETIN MSGPKT+MS.XSO ANDB EXPBOT NE #0 THEN
013000  032737  000002  002346                                                   BIT     #XO.BOT,MSGPKT+MS.XSO
013006  001404                                                                  BEQ     50167$
013010  105737  003520                                                          TSTB    EXPBOT
013014  001401                                                                  BEQ     50167$
                                                                        ;IF AT BOT AND BOT IS EXPECTED:
013016  000452                          BR TC2RTN                       ;RETURN-TCC2 CAUSED BY EXPECTED BOT.
013020                                  ENDIF
013020                                                                  50167$:
013020                                IF L$TEST EQ #3 THEN
013020  023727  002114  000003                                                   CMP     L$TEST,#3
013026  001011                                                                  BNE     50170$
013030                                 IF PCMDWD EQ #WTM AND CMDWRD EQ #SRR THEN
013030  023727  003432  100011                                                   CMP     PCMDWD,#WTM
013036  001005                                                                  BNE     50171$
013040  023727  003426  104410                                                   CMP     CMDWRD,#SRR
013046  001001                                                                  BNE     50171$
013050  000435                           BR     TC2RTN
013052                                  ENDIF
013052                                                                  50171$:
013052                                ENDIF                                     50170$:
013052                          IF #XO.RLS!XO.RLL!XO.TMK!XO.LET!XO.BOT SETIN MSGPKT+MS.XSO THEN
013052  032737  170002  002346                                                   BIT     #XO.RLS!XO.RLL!XO.TMK!XO.LET
013060  001431                                                                  BEQ     50172$
                                                                        ;IF TCC2 CAUSED BY ANYTHING BUT EOT:
013062                            IFB RANDOM EQ #0 ORB VFYFLG NE #0 THEN
013062  105737  003521                                                          TSTB    RANDOM
013066  001403                                                                  BEQ     50173$
013070  105737  003522                                                          TSTB    VFYFLG
013074  001423                                                                  BEQ     50174$
013076                                                                  50173$:
                                                                      ;IF NOT IN RANDOM OR IF CMD IS WTV:
013076                              IFB IRE EQ #0 THEN                ;IF RFC ERROR REPORTS ARE ALLOWED:
013076  105737  003526                                                          TSTB    IRE
013102  001020                                                                  BNE     50175$
013104                                IFB ERRREC NE #0 THEN           ;IF WE ARE IN ERROR RECOVERY THEN:
013104  105737  003475                                                          TSTB    ERRREC
013110  001403                                                                  BEQ     50176$
013112                                  LET UNREC :B= UNREC + #1       ;SET UNRECOVERABLE FLAG FOR LOG.
013112  105237  003474                                                          INCB    UNREC
013116                                ELSE                            ;ELSE - IF NOT IN ERROR RECOVERY:
013116  000402                                                                  BR      50177$
013120                                                                  50176$:
013120                                  LET SCCNT(R5) := SCCNT(R5) + #1 ;INCREMENT THE SPEC COND COUNTER.
013120  005265  003270                                                          INC     SCCNT(R5)
013124                                ENDIF
013124                                                                  50177$:
013124                                  LET HRDCNT(R5) := HRDCNT(R5) + #1   ;UPDATE HARD ERROR COUNT.
```

```
      013124  005265  003310                                          INC     HRDCNT(R5)
 3639 013130                                  ERRHRD 7,TSAM,STAERM    ;REPORT TAPE STATUS ALERT.
      013130  104456                                                  TRAP    C$ERHRD
      013132  000007                                                  .WORD   7
      013134  004520                                                  .WORD   TSAM
      013136  005626                                                  .WORD   STAERM
 3640 013140  004737  014050                    JSR     PC,RAMDUM     ;GO DO RAM DUMP
 3641 013144                                  ENDIF
      013144                                                          50175$:
 3642 013144                                ENDIF
      013144                                                          50174$:
 3643 013144                              ENDIF
      013144                                                          50172$:
 3644 013144  000207              TC2RTN:  RTS PC                    ;RETURN.
 3645
 3646
 3647
 3648
 3649
 3650
 3651                             ;       THE SPECIFIED FUNCTION WAS NOT INITIATED.  BITS OF INTEREST ARE
 3652                             ;       RMR, OFL, VCK, BOT, ILC, WLE, ILA, AND NBA.
 3653                             ;       INPUTS:
 3654                             ;       OUTPUTS:
 3655                             ;       REGISTERS:      R2,R4
 3656                             ;       CALLS:          DROPU
 3657
 3658 013146                      TCC3::  ERRDF 8,FUNRM,STAERM    ;REPORT FUNCTION REJECT.
      013146  104455                                                  TRAP    C$ERDF
      013150  000010                                                  .WORD   8
      013152  004435                                                  .WORD   FUNRM
      013154  005626                                                  .WORD   STAERM
 3659 013156  004737  014050              JSR     PC,RAMDUM      ;GO DO RAM DUMP
 3660 013162  004737  017402              JSR     PC,DROPU       ;DROP THE UNIT.
 3661 013166  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

013170                          TCC4::  IF DEVTBL(R5) EQ #NINUSE THEN
013170  026527  002536  177774                                        CMP     DEVTBL(R5),#NINUSE
013176  001002                                                        BNE     50200$
013200  000137  013666                        JMP     3$
013204                                  ENDIF
013204                                                              50200$:
013204  023727  003434  000002                                      CMP     CMDLG,#2
013212  001142                                                      BNE     50201$
013214  105737  002211                                              TSTB    BADTSW
013220  001537                                                      BEQ     50201$
013222                          IFB ERRREC EQ #0 ANDB ERCVER NE #0 THEN
013222  105737  003475                                              TSTB    ERRREC
013226  001011                                                      BNE     50202$
013230  105737  002207                                              TSTB    ERCVER
013234  001406                                                      BEQ     50202$
013236                          ERRSOFT 9,RERM,STAERM ;
013236  104457                                                      TRAP    C$ERSOFT
013240  000011                                                      .WORD   9
013242  004632                                                      .WORD   RERM
013244  005626                                                      .WORD   STAERM
013246  004737  014050          JSR   PC,RAMDUM             ;GO DO RAM DUMP
013252                          ENDIF
013252                                                              50202$:
013252                          IFB IREC EQ #0 THEN         ;
013252  105737  002210                                              TSTB    IREC
013256  001115                                                      BNE     50203$
013260                            LET ERRREC :B= ERRREC + #1 ;RETRY FLAG FOR EXCUTE SUBR; DON'T UPDATE REC CNT
013260  105237  003475                                              INCB    ERRREC
013264                            LET WRTYER :B= WRTYER + #1 ;REWRITE ERROR FLAG FOR WRTY SUBR
013264  105237  003470                                              INCB    WRTYER
013270                            IFB WRTYFG EQ #0 THEN     ;FIRST RETRY ON THIS RECORD; SUBSEQUENT
013270  105737  003467                                              TSTB    WRTYFG
013274  001105                                                      BNE     50204$
                                                            ;RETRIES WITH TCC4 ERRORS BY-PASS THIS SECTION
013276                              LET WTYWRD := CMDWRD    ;SAVE WRITE COMMAND PACKET
013276  013737  003426  014730                                      MOV     CMDWRD,WTYWRD
013304                              LET WTYCMD := CMDPKT    ;
013304  013737  002314  014726                                      MOV     CMDPKT,WTYCMD
```

```
3696 013312                           LET WTYBRF := CMDPKT+CP.CNT          ;
     013312  013737  002322  014732                                        MOV     CMDPKT+CP.CNT,WTYBRF
3697 013320                           LET RWERR :B= RWERR + #1   ;LOG SUBR FLAG: COUNT WRT ERRORS
     013320  105237  003473                                                INCB    RWERR
3698 013324                           LET WRTYFG :B= WRTYFG + #1        ;RETRY IN PROGRESS FLAG
     013324  105237  003467                                                INCB    WRTYFG
3699 013330                           REPEAT
     013330                                                                50205$:
3700 013330                             LET WRTYCT(R5) := WRTYCT(R5) + #1      ;COUNT GLOBAL WRITE RETRIES
     013330  005265  003250                                                INC     WRTYCT(R5)
3701 013334                             LET RETRYC := #0               ;CLEAR # OF RETRIES PER RECORD
     013334  005037  003464                                                CLR     RETRYC
3702 013340                             LET RPTCNT :B= #0              ;CLEAR # OF REPEATS
     013340  105037  003466                                                CLRB    RPTCNT
3703 013344  004737  014350             JSR PC,WRTY            ;CALL WRITE RETRY
3704 013350                                     IF DEVTBL(R5) EQ #NINUSE THEN
     013350  026527  002536  177774                                        CMP     DEVTBL(R5),#NINUSE
     013356  001001                                                        BNE     50206$
3705 013360  000542                     BR 3$
3706 013362                                     ENDIF
     013362                                                                50206$:
3707 013362                           UNTILB WRTYER EQ #0 OR @BTPT HIS #40.   ;REPEAT RETRIES ON SAME RECORD
     013362  105737  003470                                                TSTB    WRTYER
     013366  001404                                                        BEQ     50207$
     013370  027727  170122  000050                                        CMP     @BTPT,#40.
     013376  103754                                                        BLO     50205$
     013400                                                                50207$:
3708                                                           ;UNTIL RECOVERED OR 20 BAD SPOTS
3709 013400                           IF @BTPT HIS #40. THEN    ;WHEN 20 BAD SPOTS LOGGED (TWR)
     013400  027727  170112  000050                                        CMP     @BTPT,#40.
     013406  103406                                                        BLO     50210$
3710 013410  004737  015146             JSR PC,BORERS          ;ERASE BAD RECORD
3711 013414                             LET RECCNT(R5) := RECCNT(R5) - #1      ;
     013414  005365  003330                                                DEC     RECCNT(R5)
3712 013420  004737  014050             JSR     PC,RAMDUM      ;GO DO RAM DUMP
3713 013424                           ENDIF
     013424                                                                50210$:
3714 013424                           IF @BTPT HIS #512. THEN
     013424  027727  170066  001000                                        CMP     @BTPT,#512.
     013432  103417                                                        BLO     50211$
3715 013434                             PRINTB  #BTMSG2        ;BAD TAPE OVERFLOW MESSAGE
     013434  012746  015021                                                        MOV     #BTMSG2,-(SP)
     013440  012746  000001                                                        MOV     #1,-(SP)
     013444  010600                                                                MOV     SP,R0
     013446  104414                                                                TRAP    C$PNTB
     013450  062706  000004                                                        ADD     #4,SP
3716 013454  004737  017402             JSR     PC DROPU       ;DROP THE UNIT.
3717 013460                             LET RECCNT(R5) :=  #0  ;
     013460  005065  003330                                                CLR     RECCNT(R5)
3718 013464                             LET @TSDB(R5) := #RWCPK ;REWIND UNIT
     013464  012775  002334  002456                                        MOV     #RWCPK,@TSDB(R5)
3719 013472                           ENDIF
     013472                                                                50211$:
3720 013472                           LET WRTYFG :B= #0         ;RETRY COMPLETE FLAG
     013472  105037  003467                                                CLRB    WRTYFG
3721 013476                           LET MISCFG :B= MISCFG + #1        ;DO NOT HALT ON THIS CMD FLG
     013476  105237  003537                                                INCB    MISCFG
```

```
3722 013502                                    LET PCMDWD :* WTYWRD       ;RESTORE ORIGINAL WRT CMD AFTER RECOVERY
     013502  013737  014730  003432                                              MOV     WTYWRD,PCMDWD
3723 013510                                    ENDIF
     013510                                                                      50204$:
3724 013510                                  ELSE
     013510  000402                                                             BR      50212$
     013512                                                                      50203$:
3725 013512                                    LET UNREC :B= UNREC + #1    ;
     013512  105237  003474                                                    INCB    UNREC
3726 013516                                    ENDIF
     013516                                                                      50212$:
3727 013516                                ELSE
     013516  000463                                                             BR      50213$
     013520                                                                      50201$:
3728 013520  004737  014212                  JSR PC,RTLE                ;CHECK FOR RETRY LIMIT EXCEEDED.
3729 013524                                  IF CMDLG GT #2 THEN        ;IF READ CMD THEN:
     013524  023727  003434  000002                                              CMP     CMDLG,#2
     013532  003411                                                             BLE     50214$
3730 013534                                    LET R2 := #RRECL SHIFT -1  ;R2=READ RETRY COUNT LIMIT / 2
     013534  012702  000020                                                     MOV     #RRECL,R2
     013540  006202                                                             ASR     R2
3731 013542                                    IF RETRYC GE R2 THEN                ;IF RETRY COUNT IS MORE THAN HALF LIMIT:
     013542  023702  003464                                                     CMP     RETRYC,R2
     013546  002403                                                             BLT     50215$
3732 013550                                      LET CMDPKT :* CMDPKT SET.BY #OPP.C        ;SET OPPOSITE BIT FOR RETRY2.
     013550  052737  020000  002314                                              BIS     #OPP.C,CMDPKT
3733 013556                                    ENDIF
     013556                                                                      50215$:
3734 013556                                  ENDIF
     013556                                                                      50214$:
3735 013556  005737  003464                IF RETRYC EQ #0 ANDB ERCVER NE #0 THEN  ;IF THIS IS THE ORIGINAL ERROR THEN:
     013562  001011                                                             TST     RETRYC
     013564  105737  002207                                                    BNE     50216$
     013570  001406                                                             TSTB    ERCVER
3736 013572                                                                     BEQ     50216$
     013572  104457                        ERRSOFT 9,RERM,STAERM       ;REPORT RECOVERABLE ERROR
     013574  000011                                                             TRAP    C$ERSOFT
     013576  004632                                                             .WORD   9
     013600  005626                                                             .WORD   RERM
3737 013602  004737  014050                                                     .WORD   STAERM
3738 013606                                JSR   PC,RAMDUM             ;GO DO RAM DUMP
     013606                                ENDIF                       ;PROVIDED OPERATOR HAS ENABLED THE REPORT
3739 013606                                                                     50216$:
     013606  005237  003464                LET RETRYC :* RETRYC + #1    ;UPDATE RETRY COUNT.
3740 013612                                                                     INC     RETRYC
     013612  052737  001000  002314        LET CMDPKT :* CMDPKT SET.BY #MOD.C1   ;SET RETRY BIT IN CMD PACKET.
3741 013620                                                                     BIS     #MOD.C1,CMDPKT
     013620  105737  002210                IFB IREC EQ #0 THEN         ;IF ERROR RECOVERY ENABLED:
     013624  001016                                                             TSTB    IREC
3742 013626                                                                     BNE     50217$
     013626  026527  002536  177774                IF DEVTBL(R5) EQ #NINUSE THEN
     013634  001001                                                             CMP     DEVTBL(R5),#NINUSE
3743 013636  000413                                                             BNE     50220$
3744 013640                                      BR 3$
     013640                                      ENDIF                                  50220$:
3745 013640                                  LET ERRREC :B= ERRREC + #1  ;SET ERROR RECOVERY FLAG.
```

```
      013640  105237  003475                                                    INCB    ERRREC
 3746 013644                            POP R2,R2                       ;POP 2 RTN ADRS FROM STACK.
      013644  012602                                                            MOV     (SP)+,R2
      013646  012602                                                            MOV     (SP)+,R2
 3747 013650  004737  010654                    JSR PC,EXCUTE                       ;GO EXECUTE THE RETRY COMMAND.
 3748 013654  000137  011274                    JMP GOWAIT                      ;GO WAIT FOR INTERRUPT + CHECK STATUS.
 3749 013660                                  ELSE                              ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
      013660  000402                                                            BR      50221$
      013662                                                                    50217$:
 3750 013662                                    LET UNREC :B= UNREC + #1        ;SET UNRECOVERABLE ERROR FLAG.
      013662  105237  003474                                                    INCB    UNREC
 3751 013666                                  ENDIF
      013666                                                                    50221$:
 3752 013666                                ENDIF
      013666                                                                    50213$:
 3753 013666  000207                3$:     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.

013670  004737  014212        TCC5::  JSR PC,RTLE                     ;CHECK FOR RETRY LIMIT EXCEEDED
                                        IF RETRYC EQ #0 THEN            ;IF THIS IS THE ORIGINAL ERROR THEN:
013674                                                                          TST     RETRYC
013674  005737  003464                                                          BNE     50222$
013700  001006
                                          ERRSOFT 10,RERM,STAERM        ;REPORT RECOVERABLE ERROR.
013702                                                                          TRAP    C$ERSOFT
013702  104457                                                                  .WORD   10
013704  000012                                                                  .WORD   RERM
013706  004632                                                                  .WORD   STAERM
013710  005626
013712  004737  014050                  JSR     PC,RAMDUM               ;GO DO RAM DUMP
                                        ENDIF
013716                                                                  50222$:
013716                                  LET RETRYC := RETRYC + #1        ;UPDATE RETRY COUNTER.
013716  005237  003464                                                          INC     RETRYC
                                        IFB IREC EQ #0 THEN             ;IF ERROR RECOVERY IS ENABLED:
013722  105737  002210                                                          TSTB    IREC
013726  001016                                                                  BNE     50223$
                                          LET ERRREC :B= ERRREC + #1     ;SET ERROR RECOVERY FLAG.
013730  105237  003475                                                          INCB    ERRREC
                                          LET RECCNT(R5) := RECCNT(R5) + #1      ;UPDATE REC COUNT
013734  005265  003330                                                          INC     RECCNT(R5)
                                          LET @DATAWT := RECCNT(R5)             ;AND INSERT IT INTO WRT BFR
013740  016577  003330  167446                                                  MOV     RECCNT(R5),@DATAWT
                                          POP R2,R2                     ;POP 2 RTN ADRS FROM STACK.
013746  012602                                                                  MOV     (SP)+,R2
013750  012602                                                                  MOV     (SP)+,R2
013752  004737  010654                    JSR PC,EXCUTE                 ;GO RE-ISSUE THE COMMAND.
013756  000137  011274                    JMP GOWAIT                    ;GO WAIT FOR INTERRUPT + CHECK STATUS.
                                        ELSE                            ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
013762                                                                          BR      50224$
013762  000402
013764                                                                  50223$:
                                          LET UNREC :B= UNREC + #1       ;SET UNRECOVERABLE ERROR FLAG.
013764  105237  003474                                                          INCB    UNREC
                                        ENDIF
013770                                                                  50224$:
013770  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

013772                          TCC6::  LET @TSDB(R5) := #RWCPK         ;ISSUE A REWIND COMMAND.
013772  012775  002334  002456                                                   MOV     #RWCPK,@TSDB(R5)
014000  004737  011672                  JSR PC,WSSR                     ;WAIT FOR SUBSYSTEM READY.
014004                                  ERRUF 11,URERM,STAERM           ;REPORT UNRECOVERABLE ERROR.
014004  104455                                                                  TRAP    C$ERDF
014006  000013                                                                  .WORD   11
014010  004654                                                                  .WORD   URERM
014012  005626                                                                  .WORD   STAERM
014014  004737  014050                  JSR     PC,RAMDUM               ;GO DO RAM DUMP
014020  004737  017402                  JSR     PC,DROPU                ;DROP THE UNIT.
014024  000207                          RTS PC                          ;RETURN
```

```
3801                                    ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 7, FATAL SUBSYSTEM
3802                                    ;       ERROR.  THE SUBSYSTEM IS INCAPABLE OF PROPERLY PERFORMING
3803                                    ;       COMMANDS OR AT LEAST ITS INTEGRITY IS SERIOUSLY QUESTIONABLE.
3804                                    ;       REFER TO THE FATAL CLASS CODE FIELD IN THE TSSR REGISTER FOR
3805                                    ;       ADDITIONAL INFORMATION ON THE TYPE OF FATAL ERROR.
3806                                    ;       INPUTS:
3807                                    ;       OUTPUTS:
3808                                    ;       REGISTERS:      R2, R4
3809                                    ;       CALLS:
3810
3811 014026                            TCC7::  ERRDF 12,FATSM,STAERM           ;REPORT FATAL SUBSYSTEM ERROR.
     014026  104455                                                                     TRAP    C$ERDF
     014030  000014                                                                     .WORD   12
     014032  004455                                                                     .WORD   FATSM
     014034  005626                                                                     .WORD   STAERM
3812 014036  004737  014050                    JSR     PC,RAMDUM               ;GO DO RAM DUMP
3813 014042  004737  017402                    JSR     PC,DROPU                ;DROP THE UNIT.
3814 014046  000207                            RTS PC                          ;RETURN.
3815
3816
3817
3818
3819 014050                            RAMDUM::
3820 014050                                  IFB RAMWRT NE #0 THEN
     014050  105737  002214                                                           TSTB    RAMWRT
     014054  001455                                                                   BEQ     50225$
3821 014056                                    PRINTX  #RAMFHR
     014056  012746  005306                                                           MOV     #RAMFHR,-(SP)
     014062  012746  000001                                                           MOV     #1,-(SP)
     014066  010600                                                                   MOV     SP,R0
     014070  104415                                                                   TRAP    C$PNTX
     014072  062706  000004                                                           ADD     #4,SP
3822 014076  012737  000010  003406                MOV     #8.,RAMSIZ      ;RAM FIELD IS 8 BYTES LONG
3823 014104  012737  000020  003342                MOV     #20,RAMHLD      ;COMMAND PACKET ADDRESS
3824 014112  004737  015736                        JSR     PC,RAMER        ;READ AND PRINT THEM
3825 014116  012737  000200  003342                MOV     #200,RAMHLD     ;CHARACTERISTICS PACKET ADDRESS
3826 014124  004737  015736                        JSR     PC,RAMER        ;READ AND PRINT THEM
3827 014130  012737  000016  003406                MOV     #14.,RAMSIZ     ;RAM FIELD IS 8 BYTES LONG
3828 014136  012737  000214  003342                MOV     #214,RAMHLD     ;MESSAGE PACKET ADDRESS
3829 014144  004737  015736                        JSR     PC,RAMER        ;READ AND PRINT THEM
3830 014150  012737  000020  003406                MOV     #16.,RAMSIZ     ;RAM FIELD IS SIXTEEN BYTES LONG
3831 014156  012737  000060  003342                MOV     #60,RAMHLD      ;SENCE BYTES ADDRESS
3832 014164  004737  015736                        JSR     PC,RAMER        ;READ AND PRINT THEM
3833 014170                                    PRINTX  #RAMLIN
     014170  012746  005450                                                           MOV     #RAMLIN,-(SP)
     014174  012746  000001                                                           MOV     #1,-(SP)
     014200  010600                                                                   MOV     SP,R0
     014202  104415                                                                   TRAP    C$PNTX
     014204  062706  000004                                                           ADD     #4,SP
3834 014210                                  ENDIF
     014210                                                                 50225$:
3835 014210  000207                              RTS     PC
3836
```

```
3838                                    ;       SUBROUTINE TO CHECK FOR RETRY LIMIT EXCEEDED.  PRINTS ERROR MESSAGE
3839                                    ;       IF EXCEEDED AND DROP UNIT UNLESS COMMAND IS A READ.
3840                                    ;       INPUTS:
3841                                    ;       OUTPUTS:
3842                                    ;       REGISTERS:      R2, R4.
3843                                    ;       CALLS:          DROPU
3844
3845
3846 014212                            RTLE::  IF CMDLG EQ #0 THEN             ;IF CMD IS NOT A READ OR WRITE THEN:
     014212  005737  003434                                                             TST     CMDLG
     014216  001012                                                             BNE     50226$
3847 014220                                    ERRDF 11,URERM,STAERM           ;REPORT UNRECOVERABLE ERROR.
     014220  104455                                                             TRAP    C$ERDF
     014222  000013                                                             .WORD   11
     014224  004654                                                             .WORD   URERM
     014226  005626                                                             .WORD   STAERM
3848 014230  004737  014050                    JSR     PC,RAMDUM               ;GO DO RAM DUMP
3849 014234  004737  017402                    JSR     PC,DROPU                ;DROP THE UNIT.
3850 014240                                    POP R2
     014240  012602                                                             MOV     (SP)+,R2
3851 014242  000441                            BR RTLRTN                       ;AND RETURN.
3852 014244                                    ENDIF
     014244                                                                     50226$:
3853 014244                                    LET RWERR :B= RWERR + #1        ;SET READ/WRITE ERROR FLAG.
     014244  105237  003473                                                     INCB    RWERR
3854 014250                                    IF CMDLG EQ #2 THEN             ;IF CMD IS A WRT OR WTM:
     014250  023727  003434  000002                                             CMP     CMDLG,#2
     014256  001020                                                             BNE     50227$
3855 014260                                      IF RETRYC EQ #WRECL THEN      ;IF RETRY COUNT HAS REACHED LIMIT:
     014260  023727  003464  000020                                             CMP     RETRYC,#WRECL
     014266  001013                                                             BNE     50230$
3856 014270                                       LET UNREC :B= UNREC + #1     ;SET UNRECOVERABLE FLAG
     014270  105237  003474                                                     INCB    UNREC
3857 014274                                       ERRDF 14,RLEXM,STAERM        ;REPORT RETRY LIMIT EXCEEDED.
     014274  104455                                                             TRAP    C$ERDF
     014276  000016                                                             .WORD   14
     014300  004372                                                             .WORD   RLEXM
     014302  005626                                                             .WORD   STAERM
3858 014304  004737  014050                       JSR  PC,RAMDUM               ;GO DO RAM DUMP
3859 014310  004737  017402                       JSR  PC,DROPU                ;DROP THE UNIT.
3860 014314                                       POP  R2
     014314  012602                                                             MOV     (SP)+,R2
3861 014316                                     ENDIF
     014316                                                                     50230$:
3862 014316                                   ELSE                              ;ELSE - CMD IS A READ:
     014316  000413                                                             BR      50231$
     014320                                                                     50227$:
3863 014320                                     IF RETRYC EQ #RRECL THEN        ;IF RETRY COUNT HAS REACHED LIMIT:
     014320  023727  003464  000020                                             CMP     RETRYC,#RRECL
     014326  001007                                                             BNE     50232$
3864 014330                                       LET UNREC :B= UNREC + #1     ;SET UNRECOVERABLE FLAG
     014330  105237  003474                                                     INCB    UNREC
3865 014334                                       ERRHRD 14,RLEXM,STAERM       ;REPORT RECOVERABLE ERROR.
     014334  104456                                                             TRAP    C$ERHRD
     014336  000016                                                             .WORD   14
     014340  004372                                                             .WORD   RLEXM
     014342  005626                                                             .WORD   STAERM
```

```
3866 014344                                  POP R2                                  MOV     (SP)+,R2
     014344  012602
3867 014346                                ENDIF                                     50232$:
     014346
3868 014346                               ENDIF                                      50231$:
     014346
3869 014346  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
                                        WRTY:: IF DEVTBL(R5) NE #NINUSE THEN            ;IF DRIVE NOT DROPPED
014350  026527  002536  177774                                          CMP     DEVTBL(R5),#NINUSE
014356  001537                                                          BEQ     50233$
                                            BEGIN RETRY
                                              REPEAT
014360                                                                  50235$:
                                                BEGIN REPEAT
                                                  REPEAT
014360                                                                  50237$:
014360  004737  015146                            JSR PC,BORERS       ;BACKSPACE/ERASE ONE RECORD
014364  105037  003470                            LET WRTYER :B= #0   ;CLEAR WRITE RETRY ERROR
                                                                        CLRB    WRTYER
014370  004737  015322                        JSR PC,REWRT            ;REWRITE RECORD ON SAME SPOT
                                                IF DEVTBL(R5) EQ #NINUSE THEN
014374  026527  002536  177774                                          CMP     DEVTBL(R5),#NINUSE
014402  001003                                                          BNE     50240$
                                                LET RPTCNT :B= #3
014404  112737  000003  003466                                          MOVB    #3,RPTCNT
                                                ENDIF
014412                                                                  50240$:
                                                LET RPTCNT :B= RPTCNT + #1  ;COUNT REPEATS
014412  105237  003466                                                  INCB    RPTCNT
                                              UNTILB RPTCNT EQ #4 ORB WRTYER NE #0  ;LIMIT: 4 REPEATS OR RECOVERED
014416  123727  003466  000004                                          CMPB    RPTCNT,#4
014424  001405                                                          BEQ     50241$
014426  105737  003470                                                  TSTB    WRTYER
014432  001752                                                          BEQ     50237$
014434                                                                  50241$:
                                            END REPEAT                  ;
014434                                                                  50236$:
                                            LET RETRYC := RETRYC + #1    ;COUNT RETRIES
014434  005237  003464                                                  INC     RETRYC
                                                IF DEVTBL(R5) EQ #NINUSE THEN
014440  026527  002536  177774                                          CMP     DEVTBL(R5),#NINUSE
014446  001002                                                          BNE     50242$
                                                BR      3$
014450  000502                                                          BR      50243$
                                                ELSE
014452                                                                  50242$:
014452  000400
014454                                                                  50243$:
                                                ENDIF
014454
```

```
3906 014454                                     IFB WRTYER EQ #0 THEN                   ;
     014454  105737  003470                                                                  TSTB    WRTYER
     014460  001001                                                                          BNE     50244$
3907 014462                                       LEAVE RETRY                           ;EXIT RETRY LOOP IF RECOVERED
     014462  000457                                                                          BR      50234$
3908 014464                                     ELSE                                    ;
     014464                                                                                  50244$:
3909 014464                                       IFB ERCVER NE #0 THEN                 ;
     014464  105737  002207                                                                  TSTB    ERCVER
     014470  001415                                                                          BEQ     50246$
3910 014472                                         PRINTB #BTMSG1,RETRYC,<B,RPTCNT>         ;PRINT SUSPECTED BA[illegible]
     014472  005046                                                                                  CLR     -(S[illegible]
     014474  153716  003466                                                                          BISB    RPT[illegible]
     014500  013746  003464                                                                          MOV     RET[illegible]
     014504  012746  014734                                                                          MOV     #BT[illegible](SP)
     014510  012746  000003                                                                          MOV     #3,-(SP)
     014514  010600                                                                                  MOV     SP,R0
     014516  104414                                                                                  TRAP    C$PNTB
     014520  062706  000010                                                                          ADD     #10,SP
3911 014524                                       ENDIF                                 ;
     014524                                                                                  50246$:
3912 014524                                       IF RETRYC EQ #1 THEN        ;ON FIRST RETRY, LOGG BAD SPOT
     014524  023727  003464  000001                                                          CMP     RETRYC,#1
     014532  001021                                                                          BNE     50247$
3913 014534                                         LET BTPT := BTADDR(R5)              ;BTPT IS BOTH THE BAD SPOT COUNTER
     014534  016537  002550  003516                                                          MOV     BTADDR(R5),BTPT
3914 014542                                         LET R4 := @BTPT + #2                ;AND THE LOGGING INDEX
     014542  017704  166750                                                                  MOV     @BTPT,R4
     014546  062704  000002                                                                  ADD     #2,R4
3915 014552                                         LET @BTPT := R4                     ;
     014552  010477  166740                                                                  MOV     R4,@BTPT
3916 014556                                         IF R4 LOS #40. THEN                 ;
     014556  020427  000050                                                                  CMP     R4,#40.
     014562  101005                                                                          BHI     50250$
3917 014564                                           LET R3 := BTPT                    ;STORE FIRST 20 BAD SPOTS
     014564  013703  003516                                                                  MOV     BTPT,R3
3918 014570                                           LET R4 := R4 + R3                 ;
     014570  060304                                                                          ADD     R3,R4
3919 014572                                           LET (R4) := RECCNT(R5)            ;
     014572  016514  003330                                                                  MOV     RECCNT(R5),(R4)
3920 014576                                         ENDIF
     014576                                                                                  50250$:
3921 014576                                       ENDIF                                 ;
     014576                                                                                  50247$:
3922 014576                                       LET ERSFLG :B= ERSFLG + #1            ;ERASE FLAG TO ERASE BAD RECORD
     014576  105237  003533                                                                  INCB    ERSFLG
3923 014602                                       LET RWERR :B= #0                      ;CANCELL "LOG" ERROR FLAG ON FAILING RETRY
     014602  105037  003473                                                                  CLRB    RWERR
3924 014606                                       LET RPTCNT :B= #0                     ;CLEAR REPEAT COUNT FOR NEXT RETRY
     014606  105037  003466                                                                  CLRB    RPTCNT
3925 014612                                     ENDIF                                   ;
     014612                                                                                  50245$:
3926 014612                                   UNTIL RETRYC EQ #4                        ;LIMIT: 4 RETRIES
     014612  023727  003464  000004                                                          CMP     RETRYC,#4
     014620  001257                                                                          BNE     50235$
3927 014622                                 END RETRY                                   ;
     014622                                                                                  50234$:
```

```
3928 014622                             IFB WRTYER NE #0 THEN                   ;
     014622  105737  003470                                                       TSTB    WRTYER
     014626  001413                                                             BEQ     50251$
3929 014630                               IFB ERCVER NE #0 THEN                 ;
     014630  105737  002207                                                       TSTB    ERCVER
     014634  001410                                                             BEQ     50252$
3930 014636                                 PRINTB #BTMSG3                      ;PRINT RETRY FAILED
     014636  012746  015076                                                       MOV     #BTMSG3,-(SP)
     014642  012746  000001                                                       MOV     #1,-(SP)
     014646  010600                                                             MOV     SP,R0
     014650  104414                                                             TRAP    C$PNTB
     014652  062706  000004                                                       ADD     #4,SP
3931 014656                               ENDIF
     014656                                                                     50252$:
3932 014656                             ENDIF
     014656                                                                     50251$:
3933 014656                           ENDIF
     014656                                                                     50233$:
3934 014656                     3$:     LET -(SP) := #10.                       ;LOOP CONTROL
     014656  012746  000012                                                       MOV     #10.,-(SP)
3935 014662                             REPEAT
     014662                                                                     50253$:
3936 014662                             DELAY   250.                            ;WAIT FOR THE UNIT TO STOP
     014662  012727  000372                                                       MOV     #250.,(PC)+
     014666  000000                                                             .WORD   0
     014670  013727  002116                                                       MOV     L$DLY,(PC)+
     014674  000000                                                             .WORD   0
     014676  005367  177772                                                       DEC     -6(PC)
     014702  001375                                                             BNE     .-4
     014704  005367  177756                                                       DEC     -22(PC)
     014710  001367                                                             BNE     .-20
3937 014712                                 LET (SP) := (SP) - #1               ;ONE LESS ITERATION
     014712  005316                                                             DEC     (SP)
3938 014714                                 UNTIL (SP) EQ #0                    ;'TILL ZERO!
     014714  005716                                                             TST     (SP)
     014716  001361                                                             BNE     50253$
3939 014720                                 LET SP := SP + #2                   ;POP THE STACK
     014720  062706  000002                                                       ADD     #2,SP
3940 014724  000207                        RTS PC
3941
3942
3943
3944
3945
3946
3947 014726  000000                 WTYCMD: .WORD   0                   ;STORAGE FOR WRITE CMD WHILE RETRYING
3948 014730  000000                 WTYWRD: .WORD   0                   ;STORAGE FOR WRITE CMD WORD WHILE RETRYING
3949 014732  000000                 WTYBRF: .WORD   0                   ;STORAGE FOR WRITE BPCR WHILE RETRYING
3950
3951
3952 014734    045    101    123  BTMSG1: .ASCIZ  /%ASUSPECT BAD SPOT AFTER %D1%A RETR(, %D1%A REPEAT%N/
     014737    125    123    120
     014742    105    103    124
     014745    040    102    101
     014750    104    040    123
     014753    120    117    124
     014756    040    101    106
```

```
        014761     124     105     122
        014764     040     045     104
        014767     061     045     101
        014772     040     122     105
        014775     124     123     131
        015000     054     040     045
        015003     104     061     045
        015006     101     040     122
        015011     105     120     105
        015014     101     124     045
        015017     116     000
  3953  015021     045     116     045  BTMSG2: .ASCIZ  /%N%ABAD TAPE OVERFLOW: CHANGE CARTRIDGE!%N%N/
        015024     101     102     101
        015027     104     040     124
        015032     101     120     105
        015035     040     117     126
        015040     105     122     106
        015043     114     117     127
        015046     072     040     103
        015051     110     101     116
        015054     107     105     040
        015057     103     101     122
        015062     124     122     111
        015065     104     107     105
        015070     041     045     116
        015073     045     116     000
  3954  015076     045     101     122  BTMSG3: .ASCIZ  /%ARETRY FAILED ON BAD SPOT...ERASED!%N/
        015101     105     124     122
        015104     131     040     106
        015107     101     111     114
        015112     105     104     040
        015115     117     116     040
        015120     102     101     104
        015123     040     123     120
        015126     117     124     056
        015131     056     056     105
        015134     122     101     123
        015137     105     104     041
        015142     045     116     000
  3955                                .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

015146                                  BORERS:: LET PCMDWD := CMDWRD   ;SET COMMAND  TO SPACE REV
015146  013737  003426  003432                                                  MOV     CMDWRD,PCMDWD
                                                LET CMDWRD := #SRR      ;
015154  012737  104410  003426                                                  MOV     #SRR,CMDWRD
                                                LET CMDPKT := CMDWRD CLR.BY #BRF.C      ;
015162  013737  003426  002314                                                  MOV     CMDWRD,CMDPKT
015170  042737  004000  002314                                                  BIC     #BRF.C,CMDPKT
                                                LET CMDSAV := CMDPKT    ;
015176  013737  002314  003430                                                  MOV     CMDPKT,CMDSAV
                                                LET CMDPKT+CP.ADL := #1 ;
015204  012737  000001  002316                                                  MOV     #1,CMDPKT+CP.ADL
                                                LET CMDLG := #0         ;
015212  005037  003434                                                          CLR     CMDLG
015216  004737  007600                          JSR PC,CMDAC            ;
015222  004737  010654                          JSR PC,EXCUTE           ;
015226  004737  011274                          JSR PC,GOWAIT           ;
015232  004737  017706                          JSR PC,CKHAE            ;
015236                                          IFB ERSFLG NE #0 THEN   ;WHEN ERASE FLAG IS SET, DO ERASE
015236  105737  003533                                                          TSTB    ERSFLG
015242  001426                                                                  BEQ     50254$
015244                                            LET PCMDWD := CMDWRD  ;
015244  013737  003426  003432                                                  MOV     CMDWRD,PCMDWD
015252                                            LET CMDWRD := #ERS    ;
015252  012737  100411  003426                                                  MOV     #ERS,CMDWRD
015260                                            LET CMDPKT := CMDWRD  ;
015260  013737  003426  002314                                                  MOV     CMDWRD,CMDPKT
015266                                            LET CMDSAV := CMDPKT  ;
015266  013737  002314  003430                                                  MOV     CMDPKT,CMDSAV
015274  004737  007600                            JSR PC,CMDAC          ;
015300  004737  010654                            JSR PC,EXCUTE
015304  004737  011274                            JSR PC,GOWAIT
015310  004737  017706                            JSR PC,CKHAE
015314                                            LET ERSFLG :B= #0
015314  105037  003533                                                          CLRB    ERSFLG
015320                                          ENDIF
015320                                                                  50254$:
015320  000207                                  RTS PC
                                        ;       SUBR TO REWRITE A BADLY WRITTEN RECORD

015322                                  REWRT:  IF DEVTBL(R5) NE #NINUSE THEN   ;IF DRIVE NOT DROPPED
015322  026527  002536  177774                                                  CMP     DEVTBL(R5),#NINUSE
015330  001441                                                                  BEQ     50255$
015332                                          LET PCMDWD := CMDWRD    ;RESTORE WRITE COMMAND PACKET
015332  013737  003426  003432                                                  MOV     CMDWRD,PCMDWD
015340                                          LET CMDWRD := WTYWRD    ;
015340  013737  014730  003426                                                  MOV     WTYWRD,CMDWRD
015346                                          LET CMDPKT := WTYCMD    ;
015346  013737  014726  002314                                                  MOV     WTYCMD,CMDPKT
015354                                          LET CMDSAV := CMDPKT    ;
015354  013737  002314  003430                                                  MOV     CMDPKT,CMDSAV
```

```
3993 015362                          LET CMDPKT+CP.ADL := DATAWT      ;
     015362  013737  003414  002316                                    MOV     DATAWT,CMDPKT+CP.ADL
3994 015370                          LET CMDPKT+CP.CNT := WTYBRF      ;
     015370  013737  014732  002322                                    MOV     WTYBRF,CMDPKT+CP.CNT
3995 015376                          LET CMDLG := #2                  ;
     015376  012737  000002  003434                                    MOV     #2,CMDLG
3996 015404  004737  007600          JSR PC,CMDAC
3997 015410  004737  010654          JSR PC,EXCUTE                    ;RE-WRITE RECORD
3998 015414                            IF DEVTBL(R5) NE #NINUSE THEN
     015414  026527  002536  177774                                    CMP     DEVTBL(R5),#NINUSE
     015422  001404                                                    BEQ     50256$
3999 015424  004737  011274          JSR PC,GOWAIT                    ;
4000 015430  004737  017706          JSR PC,CKHAE                     ;
4001 015434                            ENDIF
     015434                                                          50256$:
4002 015434                          ENDIF
     015434                                                          50255$:
4003 015434  000207                  RTS PC
```

```
4005                                    ;       SUBROUTINE TO LOG BYTES READ/WRITTEN.
4006                                    ;       ALSO UPDATES READ/WRITE ERROR COUNTERS.
4007                                    ;       INPUTS:
4008                                    ;       OUTPUTS:
4009                                    ;       REGISTERS:      R2, R3, R4.
4010                                    ;       CALLS:
4011
4012 015436                            LOG::   IFB ERLOG EQ #0 THEN            ;IF DATA AND ERRORS HAVE NOT BEEN LOGGED THEN:
     015436  105737  003472                                                           TSTB    ERLOG
     015442  001126                                                                  BNE     50257$
4013 015444                                    LET ERLOG :B= ERLOG + #1         ;SET LOG DONE FLAG.
     015444  105237  003472                                                           INCB    ERLOG
4014 015450                                    LET R4 := CMDLG                  ;GET CURRENT CMD LOGGING CODE.
     015450  013704  003434                                                           MOV     CMDLG,R4
4015 015454                                    IF R4 NE #0 THEN                 ;IF THERE IS A CODE THEN:
     015454  005704                                                                  TST     R4
     015456  001520                                                                  BEQ     50260$
4016 015460                                      LET R4 := R4 - #2              ;ADJUST THE CODE FOR TABLE INDEX.
     015460  162704  000002                                                           SUB     #2,R4
4017 015464                                      LET R2 := R5 + BINC(R4) + #CNTBGN  ;R2 = ADR OF BYTE COUNT LSW.
     015464  010502                                                                  MOV     R5,R2
     015466  066402  015722                                                          ADD     BINC(R4),R2
     015472  062702  002560                                                          ADD     #CNTBGN,R2
4018 015476                                      LET (R2) := (R2) + BRFCNT      ;ADD BRF TO LSW.
     015476  063712  003424                                                          ADD     BRFCNT,(R2)
4019 015502                                      IF MSGPKT+MS.RFC LOS BRFCNT THEN  ;IF THE RFC IS LOWER OR THE SAME AS BRF THEN:
     015502  023737  002344  003424                                                  CMP     MSGPKT+MS.RFC,BRFCNT
     015510  101002                                                                  BHI     50261$
4020 015512
     015512  163712  002344                        LET (R2) := (R2) - MSGPKT+MS.RFC  ;SUBTRACT RFC FROM EXPECTED BRF.
                                                                                    SUB     MSGPKT+MS.RFC,(R2)
4021 015516                                    ENDIF
     015516                                                                         50261$:
4022 015516                                    LET R3 := R2 + #10               ;R3 = ADR OF 2ND WORD.
     015516  010203                                                                  MOV     R2,R3
     015520  062703  000010                                                          ADD     #10,R3
4023 015524                                    WHILE (R2) GT #999. DO
     015524                                                                         50262$:
     015524  021227  001747                                                          CMP     (R2),#999.
     015530  003404                                                                  BLE     50263$
4024 015532                                      LET (R2) := (R2) - #1000.      ;UPDATE BYTE COUNT
     015532  162712  001750                                                          SUB     #1000.,(R2)
4025 015536                                      LET (R3) := (R3) + #1          ;2ND WORD.
     015536  005213                                                                 INC     (R3)
4026 015540                                    ENDDO
     015540  000771                                                                 BR      50262$
     015542                                                                         50263$:
4027 015542                                    LET R2 := R3 + #10               ;R2 = ADR OF 3RD WORD.
     015542  010302                                                                  MOV     R3,R2
     015544  062702  000010                                                          ADD     #10,R2
4028 015550                                    WHILE (R3) GT #999. DO
     015550                                                                         50264$:
     015550  021327  001747                                                          CMP     (R3),#999.
     015554  003404                                                                  BLE     50265$
4029 015556                                      LET (R3) := (R3) - #1000.      ;UPDATE BYTE COUNT
     015556  162713  001750                                                          SUB     #1000.,(R3)
4030 015562                                      LET (R2) := (R2) + #1          ;3RD WORD.
     015562  005212                                                                 INC     (R2)
```

```
4031 015564                                ENDDO                                          BR      50264$
     015564  000771                                                                       50265$:
     015566
4032 015566                                LET R3 := R2 + #10          ;R3 = ADR OF 4TH WORD.
     015566  010203                                                                       MOV     R2,R3
     015570  062703  000010                                                               ADD     #10,R3
4033 015574                                WHILE (R2) GT #999. DO
     015574                                                                             50266$:
     015574  021227  001747                                                             CMP     (R2),#999.
     015600  003404                                                                     BLE     50267$
4034 015602                                  LET (R2) := (R2) - #1000.  ;UPDATE BYTE COUNT
     015602  162712  001750                                                             SUB     #1000.,(R2)
4035 015606                                  LET (R3) := (R3) + #1      ;4TH WORD.
     015606  005213                                                                     INC     (R3)
4036 015610                                ENDDO
     015610  000771                                                                     BR      50266$
     015612                                                                             50267$:
4037 015612                                IFB RWERR NE #0 THEN        ;IF R/W ERROR, UPDATE ERROR COUNT.
     015612  105737  003473                                                             TSTB    RWERR
     015616  001440                                                                     BEQ     50270$
4038 015620                                  LET R2 := R5 + EINC(R4) + #WRREC  ;R2 = ADR OF COUNTER.
     015620  010502                                                                     MOV     R5,R2
     015622  066402  015730                                                             ADD     EINC(R4),R2
     015626  062702  002720                                                             ADD     #WRREC,R2
4039 015632                                  IFB UNREC NE #0 THEN      ;IS THE ERROR UNRECOVERABLE?
     015632  105737  003474                                                             TSTB    UNREC
     015636  001404                                                                     BEQ     50271$
4040 015640                                    LET R2 := R2 + #10      ;YES, POINT TO NEXT COUNTER.
     015640  062702  000010                                                             ADD     #10,R2
4041 015644                                  LET (R2) := (R2) + #1     ;UPDATE THE ERROR COUNTER
     015644  005212                                                                     INC     (R2)
4042 015646                                ELSE                        ;ELSE - IF ERROR IS RECOVERABLE:
     015646  000424                                                                     BR      50272$
     015650                                                                             50271$:
4043 015650                                  LET (R2) := (R2) + #1     ;UPDATE THE ERROR COUNTER
     015650  005212                                                                     INC     (R2)
4044 015652                                  IFB IREC EQ #0 THEN       ;IF ERROR RECOVERY IS ENABLED:
     015652  105737  002210                                                             TSTB    IREC
     015656  001020                                                                     BNE     50273$
4045 015660                                    IFB DROPED EQ #0 ANDB ERCVER NE #0 THEN ;IF UNIT HAS NOT BEEN DROPPED:
     015660  105737  003527                                                             TSTB    DROPED
     015664  001015                                                                     BNE     50274$
     015666  105737  002207                                                             TSTB    ERCVER
     015672  001412                                                                     BEQ     50274$
4046 015674                                      PRINTB #NURTY1,RETRYC     ;PRINT # OF RETRIES TO RECOVER
     015674  013746  003464                                                                     MOV     RETRYC,-(SP)
     015700  012746  005155                                                                     MOV     #NURTY1,-(SP)
     015704  012746  000002                                                                     MOV     #2,-(SP)
     015710  010600                                                                             MOV     SP,R0
     015712  104414                                                                             TRAP    C$PNTB
     015714  062706  000006                                                                     ADD     #6,SP
4047 015720                                    ENDIF                     ;PROVIDED PRINT HAS BEEN ENABLED
     015720                                                                             50274$:
4048 015720                                  ENDIF
     015720                                                                             50273$:
4049 015720                                ENDIF
     015720                                                                             50272$:
```

```
4050 015720                                  ENDIF
     015720                                                                  50270$:
4051 015720                                ENDIF
     015720                                                                  50260$:
4052 015720                                ENDIF
     015720                                                                  50257$:
4053 015720  000207                        RTS PC
4054                              ;        INDEXES TO BYTE COUNTERS.
4055 015722  000000              BINC:    0                                  ;WRITE.
4056 015724  000040                       40                                 ;READ REV.
4057 015726  000100                       100                                ;READ FWD.
4058                              ;        INDEXES TO READ/WRITE ERROR COUNTERS.
4059 015730  000000              EINC:    0                                  ;WRITE.
4060 015732  000020                       20                                 ;READ REV.
4061 015734  000040                       40                                 ;READ FWD.
4062
4063
4064
```

```
                        .SBTTL RAMER - READ AND DISPLAY SELECTED RAM
                ;+
                ;ROUTINE TO READ THE SELECTED RAM LOCATIONS
                ;-
015736  010546          RAMER:: MOV     R5,-(SP)
015740  010446                  MOV     R4,-(SP)
015742  010346                  MOV     R3,-(SP)
015744  010246                  MOV     R2,-(SP)
015746  010146                  MOV     R1,-(SP)
015750  012701  003346          MOV     #RAMDATA,R1     ;ADDRESS TO SAVE THE RAM DATA
015754  013702  003342          MOV     RAMHLD,R2       ;BYTE ADDRESS OF THE FIRST RAM DATA
015760  013703  003406          MOV     RAMSIZ,R3       ;SET THE SIZE OF THE READ UP
015764  016504  002456          MOV     TSDB(R5),R4     ;MOV THE TSDB ADDRESS INTO R4
015770  005204                  INC     R4              ;ADD 1 TO IT
015772  000240          10$:    NOP
015774  004737  011672          JSR     PC,WSSR         ;WAIT FOR THE SSR TO SET
016000  110214                  MOVB    R2,(R4)         ;SELECT NEXT RAM ADDRESS
016002  004737  011672          JSR     PC,WSSR         ;WAIT FOR SSR TO SET
016006  117521  002456          MOVB    @TSBA(R5),(R1)+ ;READ THE RAM DATA
016012  062702  000001  20$:    ADD     #1,R2           ;ADDRESS OF THE NEXT RAM LOCATION
016016  077313                  SOB     R3,10$          ;NUMBER OF LOCATIONS COUNTER
016020  013704  003406          MOV     RAMSIZ,R4       ;GET THE RAM SIZE
016024  013702  003342          MOV     RAMHLD,R2       ;GET THE STARTING RAM ADDRESS
016030  060204                  ADD     R2,R4           ;CALCULATE THE END ADDRESS
016032  162704  000001          SUB     #1,R4           ;CORRECT VALUE OF PRINTOUT
016036                          PRINTX  #RAMIOP,R2,R4   ;RAM ADDRESS = 10 - 17, ETC.
016036  010446                                                  MOV     R4,-(SP)
016040  010246                                                  MOV     R2,-(SP)
016042  012746  005365                                          MOV     #RAMIOP,-(SP)
016046  012746  000003                                          MOV     #3,-(SP)
016052  010600                                                  MOV     SP,R0
016054  104415                                                  TRAP    C$PNTX
016056  062706  000010                                          ADD     #10,SP
016062  012701  003346          MOV     #RAMDATA,R1     ;ADDRESS OF WHERE RAM DATA IS
016066  013703  003406          MOV     RAMSIZ,R3       ;THE SIZE OF THE RAM FIELD READ
016072  005004          30$:    CLR     R4              ;NO EXTRA DATA LEFT OVER
016074  112104                  MOVB    (R1)+,R4        ;PICK UP BYTE OF RAM DATA
016076  042704  177400          BIC     #177400,R4      ;GET RID OF SIGN EXTEND
016102                          PRINTX  #RAMPD,R4       ;"010 211 111 222 377 000 123 134 ETC."
016102  010446                                                  MOV     R4,-(SP)
016104  012746  005436                                          MOV     #RAMPD,-(SP)
016110  012746  000002                                          MOV     #2,-(SP)
016114  010600                                                  MOV     SP,R0
016116  104415                                                  TRAP    C$PNTX
016120  062706  000006                                          ADD     #6,SP
016124  077316                  SOB     R3,30$          ;LOOP UNTIL ALL PRINTED
016126  012601                  MOV     (SP)+,R1
016130  012602                  MOV     (SP)+,R2
016132  012603                  MOV     (SP)+,R3
016134  012604                  MOV     (SP)+,R4
016136  012605                  MOV     (SP)+,R5
016140  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:
```

```
4110                                    ;       REGISTERS:
4111                                    ;       CALLS:          VFEXC.
4112
4113 016142                             VFYDAT::  LET CMPDAT := #0              ;BTL
     016142  005037  003410                                             CLR     CMPDAT
4114 016146                                       IFB VFYFLG NE #0 THEN         ;IF DATA IS TO BE VERIFIED:
     016146  105737  003522                                             TSTB    VFYFLG
     016152  001437                                                     BEQ     50275$
4115 016154                                        IFB STREAM EQ #0 THEN
     016154  105737  003532                                             TSTB    STREAM
     016160  001015                                                     BNE     50276$
4116 016162                                         LET PCMDWD := CMDWRD                ;SAVE THE PREVIOUS COMMAND WORD.
     016162  013737  003426  003432                                     MOV     CMDWRD,PCMDWD
4117 016170                                         LET CMDWRD := #RDR          ;COMMAND IS READ REV.
     016170  012737  104401  003426                                     MOV     #RDR,CMDWRD
4118 016176                                         LET CMDLG  := #4            ;SET UP CMD LOGGING INDEX.
     016176  012737  000004  003434                                     MOV     #4,CMDLG
4119 016204  004737  016254                         JSR PC,VFEXC                ;GO READ ALL THE RECORDS REV.
4120 016210                                         LET CMPDAT := #0            ;BTL
     016210  005037  003410                                             CLR     CMPDAT
4121 016214                                        ENDIF
     016214                                                             50276$:
4122 016214                                       IF DEVTBL(R5) NE #NINUSE THEN
     016214  026527  002536  177774                                     CMP     DEVTBL(R5),#NINUSE
     016222  001413                                                     BEQ     50277$
4123 016224                                         LET PCMDWD := CMDWRD          ;SAVE THE PREVIOUS COMMAND WORD.
     016224  013737  003426  003432                                     MOV     CMDWRD,PCMDWD
4124 016232                                         LET CMDWRD := #RDF          ;COMMAND IS READ FWD.
     016232  012737  104001  003426                                     MOV     #RDF,CMDWRD
4125 016240                                         LET CMDLG  := #6            ;SET UP CMD LOGGING INDEX.
     016240  012737  000006  003434                                     MOV     #6,CMDLG
4126 016246  004737  016254                         JSR   PC,VFEXC              ;GO READ ALL RECORDS FWD.
4127 016252                                        ENDIF
     016252                                                             50277$:
4128 016252                                       ENDIF
     016252                                                             50275$:
4129 016252  000207                     RTS     PC                      ;RETURN.
```

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

                                VFEXC:: LET CMDPKT := CMDWRD CLR.BY @BRF.C  ;COMMAND PACKET = READ REV OR FWD.
016254                                                                          MOV     CMDWRD,CMDPKT
016262  042737  004000  002314                                                BIC     @BRF.C,CMDPKT
                                        IFB SWBFLG NE @0 THEN           ;IF BYTES ARE TO BE SWAPPED:
016270                                                                          TSTB    SWBFLG
016274  001403                                                                  BEQ     50300$
                                          LET CMDPKT := CMDPKT SET.BY @SWB.C  ;SET SWAB BIT IN CMD PACKET.
016276  052737  010000  002314                                                BIS     @SWB.C,CMDPKT
                                        ENDIF
016304                                                                          50300$:
                                        LET CMDSAV := CMDPKT            ;SAVE COMMAND PACKET 1ST WORD.
016304  013737  002314  003430                                                MOV     CMDPKT,CMDSAV
                                        MOV     DATARD,CMDPKT+CP.ADL    ;SAVE BUFFER START ADDRESS.
                                        LET NCNT := @0                  ;CLEAR NUMBER OF OPERATIONS.
016320  005037  003420                                                        CLR     NCNT
                                        WHILE NCNT LT NCNT1 DO          ;WHILE THERE ARE RECORDS REMAINING:
016324                                                                          50301$:
016324  023737  003420  003422                                                CMP     NCNT,NCNT1
016332  002101                                                                BGE     50302$
                                          JSR PC,CMDAC                  ;STORE CMD ASCII IN ERROR MSG.
                                          TSTB  STREAM                  ;CHECK IF WE ARE STREAMING
                                          BNE   1$                      ;BRANCH OVER DEVTBL CHECK. THIS ENABLES
                                                                ;US TO TEST ONE DRIVE AT A TIME.
                                          JSR   PC,FIRSTU               ;SET UP FOR FIRST UNIT.
                                          WHILE DEVTBL(R5) NE @END DO   ;WHILE THERE ARE DEVICES REMAINING:
016352                                                                          50303$:
016352  026527  002536  177777                                                CMP     DEVTBL(R5),@END
016360  001445                                                                BEQ     50304$
                                1$:               IF @MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
016362  032737  000400  003426                                                BIT     @MOD.CO,CMDWRD
016370  001421                                                                BEQ     50305$
                                            IF @XO.BOT NOTSETIN EOTFLG(R5) THEN ;IF NOT AT BOT
016372  032765  000002  003506                                                BIT     @XO.BOT,EOTFLG(R5)
016400  001014                                                                BNE     50306$
                                              IF @XO.EOT SETIN EOTFLG(R5) THEN  ;BUT IF AT EOT
016402  032765  000001  003506                                                BIT     @XO.EOT,EOTFLG(R5)
016410  001406                                                                BEQ     50307$
                                                IFB ALLEOT NE @0 THEN           ;AND ALL OTHERS AT EOT
016412  105737  003531                                                        TSTB    ALLEOT
016416  001402                                                                BEQ     50310$
                                                  JSR PC,VFISU                  ;THEN READ VERIFY
                                                ENDIF                           ;IF NOT ALL AT EOT, FREEZE UNIT(S) AT E
016424                                                                          50310$:
                                              ELSE                              ;IF NOT AT BOT AND
016424  000402                                                                  BR      50311$
016426                                                                          50307$:
                                                JSR PC,VFISU                    ;NOT AT EOT, READ VFY
                                              ENDIF
016432                                                                          50311$:
                                            ENDIF
016432                                                                          50306$:
```

```
4161 016432                                        ELSE                          ;ELSE IF CMD IS NOT REVERSE:
     016432  000412                                                                BR      50312$
     016434                                                                       50305$:
4162 016434                                          IF @X0.EOT NOTSETIN EOTFLG(R5) OR @CMD.CO NOTSETIN CMDWRD THEN
     016434  032765  000001  003506                                               BIT     @X0.EOT,EOTFLG(R5)
     016442  001404                                                               BEQ     50313$
     016444  032737  000001  003426                                               BIT     @CMD.CO,CMDWRD
     016452  001002                                                               BNE     50314$
     016454                                                                       50313$:
4163                                                                             ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
4164 016454  004737  016540                          JSR PC,VFISU                  ;ISSUE CMD, CHECK STATUS AND DATA.
4165 016460                                          ENDIF
     016460                                                                       50314$:
4166 016460                                          ENDIF
     016460                                                                       50312$:
4167 016460  105737  003532                        TSTB STREAM                   ;CHECK FOR TEST OF ON UNIT AT A TIME.
4168 016464  001003                                BNE 2$                        ;BRANCH, IF STREAMING TESTS.
4169 016466  004737  017346                        JSR PC,NEXTU                  ;GO FIND THE NEXT UNIT.
4170 016472                                        ENDDO
     016472  000727                                                               BR      50303$
     016474                                                                       50304$:
4171 016474  004737  017706          2$:           JSR   PC,CKHAE                ;CHECK FOR HALT AFTER EACH CMD.
4172 016500                                        IF DEVTBL(R5) EQ @NINUSE THEN ;IF DRIVES BEEN DROPPED EXIT
     016500  026527  002536  177774                                               CMP     DEVTBL(R5),@NINUSE
     016506  001005                                                               BNE     50315$
4173 016510                                        LET NCNT := NCNT1 - @1
     016510  013737  003422  003420                                               MOV     NCNT1,NCNT
     016516  005337  003420                                                       DEC     NCNT
4174 016522                                        ENDIF
     016522                                                                       50315$:
4175 016522                                          LET NCNT := NCNT + @1       ;UPDATE THE RECORD COUNT.
     016522  005237  003420                                                       INC     NCNT
4176 016526                                          LET PCMDWD := CMDWRD        ;SAVE PREVIOUS COMMAND WORD.
     016526  013737  003426  003432                                               MOV     CMDWRD,PCMDWD
4177
4178 016534                                        ENDDO
     016534  000673                                                               BR      50301$
     016536                                                                       50302$:
4179 016536  000207                                RTS     PC                    ;RETURN.
```

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

                                VFISU::
                                    LET R2 := DATARD + #8.      ;INIT READ BUFFER POINTER.
016540  013702  003416                                                  MOV     DATARD,R2
016544  062702  000010                                                  ADD     #8.,R2
                                    WHILE R2 NE DATARD DO       ;UNTIL 8 BYTES HAVE BEEN SET,
016550                                                                  50316$:
016550  020237  003416                                                  CMP     R2,DATARD
016554  001403                                                          BEQ     50317$
                                    LET -(R2) := #-1            ;INIT READ BUFFER.
016556  012742  177777                                                  MOV     #-1,-(R2)
                                    ENDDO
016562  000772                                                          BR      50316$
016564                                                                  50317$:
016564  004737  010654              JSR PC,EXCUTE               ;GO EXECUTE THE COMMAND.
                                    IFB DROPED EQ #0 THEN       ;IF UNIT HAS NOT BEEN DROPPED THEN:
016570  105737  003527                                                  TSTB    DROPED
016574  001002                                                          BNE     50320$
016576  004737  011274              JSR PC,GOWAIT       ;GO WAIT FOR DONE BIT.
                                    ENDIF
016602                                                                  50320$:
                                    IFB DROPED EQ #0 THEN       ;IF UNIT HAS NOT BEEN DROPPED THEN:
016602  105737  003527                                                  TSTB    DROPED
016606  001025                                                          BNE     50321$
                                      IF #XO.BOT NOTSETIN EOTFLG(R5) THEN       ;WHEN NOT REVERSED INTO BOT, THEN
016610  032765  000002  003506                                          BIT     #XC.BOT,EOTFLG(R5)
016616  001021                                                          BNE     50322$
                                        IF L$TEST NE #3 THEN
016620  023727  002114  000003                                          CMP     L$TEST,#3
016626  001403                                                          BEQ     50323$
                                          JSR   PC,CKDATA       ;GO VERIFY DATA.
                                          ELSE
016634  000412                                                          BR      50324$
016636                                                                  50323$:
                                          INC   CMPDAT          ;ONE MORE XFER BEFORE A COMPARISON
                                          IF CMPDAT GT DATRAT THEN
016642  023737  003410  003412                                          CMP     CMPDAT,DATRAT
016650  003404                                                          BLE     50325$
                                            JSR         PC,CKDATA
                                            LET CMPDAT := #0
016656  005037  003410                                                  CLR     CMPDAT
                                          ENDIF
016662                                                                  50325$:
                                        ENDIF
016662                                                                  50324$:
                                      ENDIF
016662                                                                  50322$:
                                    ENDIF
016662                                                                  50321$:
                                    RTS PC
```

```
                        ;       SUBROUTINE TO COMPARE DATA BETWEEN READ AND WRITE BUFFERS
                        ;       AND PRINT ERROR MESSAGE ON MISCOMPARE.
                        ;       INPUTS:
                        ;       OUTPUTS:
                        ;       REGISTERS:      R2, R3, R4.
                        ;       CALLS:          GCMDA
                        
016664                  CKDATA:: LET R3 := BRFCNT - MSGPKT+MS.RFC ;COMPUTE REC LENGTH READ
016664  013703  003424                                                  MOV     BRFCNT,R3
016670  163703  002344                                                  SUB     MSGPKT+MS.RFC,R3
016674                          IF R3 EQ #0 THEN                ;WHEN NO DATA RECEIVED
016674  005703                                                          TST     R3
016676  001015                                                          BNE     50326$
016700                            ERRHRD 17,WTVERM,DTAERM               ;PRINT ERROR AND EXIT
016700  104456                                                          TRAP    C$ERHRD
016702  000021                                                          .WORD   17
016704  004246                                                          .WORD   WTVERM
016706  005460                                                          .WORD   DTAERM
016710                            PRINTB #DTAER4                ;COMPARE ROUTINE
016710  012746  005072                                                  MOV     #DTAER4,-(SP)
016714  012746  000001                                                  MOV     #1,-(SP)
016720  010600                                                          MOV     SP,R0
016722  104414                                                          TRAP    C$PNTB
016724  062706  000004                                                  ADD     #4,SP
016730                          ELSE
016730  000560                                                          BR      50327$
016732                                                                  50326$:
016732                  IF R3 HI BRFCNT THEN            ;WHEN REC READ IS LONGER
016732  020337  003424                                                  CMP     R3,BRFCNT
016736  101417                                                          BLOS    50330$
016740                            ERRHRD 17,WTVERM,DTAERM               ;THAN EXPECTED, PRINT
016740  104456                                                          TRAP    C$ERHRD
016742  000021                                                          .WORD   17
016744  004246                                                          .WORD   WTVERM
016746  005460                                                          .WORD   DTAERM
016750                            PRINTB #DTAER5,CMDPKT+CP.CNT          ;AN ERROR MESSAGE
016750  013746  002322                                                  MOV     CMDPKT+CP.CNT,-(SP)
016754  012746  005113                                                  MOV     #DTAER5,-(SP)
016760  012746  000002                                                  MOV     #2,-(SP)
016764  010600                                                          MOV     SP,R0
016766  104414                                                          TRAP    C$PNTB
016770  062706  000006                                                  ADD     #6,SP
016774                  ELSE                            ;AND EXIT ROUTINE
016774  000536                                                          BR      50331$
016776                                                                  50330$:
016776                    LET CKDCNT := R3 - #1                 ;SAVE VERIFICATION LENGTH - 1.
016776  010337  017274                                                  MOV     R3,CKDCNT
017002  005337  017274                                                  DEC     CKDCNT
017006  005037  017276            CLR CKDFF                     ;CLEAR # OF BYTES IN ERROR COUNTER.
017012  005002                    CLR R2                        ;INIT BYTE COUNTER
017014                            LET R3 := DATAWT              ;GET WRITE BUFFER ADDRESS.
017014  013703  003414                                                  MOV     DATAWT,R3
017020                            LET R4 := DATARD              ;GET READ BUFFER ADDRESS.
017020  013704  003416                                                  MOV     DATARD,R4
017024                            IFB T1SWB NE #0 THEN          ;WHEN RUNNING TEST1-SUB 12.
017024  105737  003530                                                  TSTB    T1SWB
017030  001401                                                          BEQ     50332$
```

```
4235 017032  000313                                  SWAB (R3)                       ;SWAP FIRST WORD OF WRT BFR
4236 017034                                          ENDIF                           ;WHICH CONTAINS THE RECORD COUNT
     017034                                                                          50332$:
4237 017034                                          REPEAT                          ;REPEAT UNTIL ALL DATA IS COMPARED:
     017034                                                                          50333$:
4238 017034                                            IF R2 EQ CKDCNT THEN          ;IF THIS IS THE LAST BYTE THEN:
     017034  020237  017274                                                         CMP     R2,CKDCNT
     017040  001011                                                                 BNE     50334$
4239 017042                                              IFB SWBFLG NE #0 THEN       ;IF BYTE SWAPPING IS ENABLED THEN:
     017042  105737  003524                                                         TSTB    SWBFLG
     017046  001406                                                                 BEQ     50335$
4240 017050                                                IF #BIT00 NOTSETIN CKDCNT THEN  ;IF RECORD LENGTH IS ODD THEN:
     017050  032737  000001  017274                                                 BIT     #BIT00,CKDCNT
     017056  001002                                                                 BNE     50336$
4241 017060  105723                                          TSTB        (R3)+      ;LAST BYTE WILL BE IN
4242 017062  105724                                          TSTB        (R4)+      ;THE UPPER BYTE.
4243 017064                                                ENDIF
     017064                                                                          50336$:
4244 017064                                              ENDIF
     017064                                                                          50335$:
4245 017064                                            ENDIF
     017064                                                                          50334$:
4246 017064  121314                                    CMPB (R3),(R4)              ;ARE THEY EQUAL.
4247 017066  001452                                    BEQ       3$                ;BR IF SO.
4248 017070  005737  017276                            TST CKDFF                   ;1 ST TIME THRU?
4249 017074  001010                                    BNE       2$                ;BR IF NOT.
4250 017076  005265  003300                            INC       VFYCNT(R5)        ;INC THE VERIFY ERROR COUNTER.
4251 017102  005265  003310                            INC HRDCNT(R5)              ;INC THE HARD ERROR COUNT.
4252 017106                                            ERRHRD 17,WTVERM,DTAERM     ;REPORT WRITE/VERIFY ERROR.
     017106  104456                                                                 TRAP    C$ERHRD
     017110  000021                                                                 .WORD   17
     017112  004246                                                                 .WORD   WTVERM
     017114  005460                                                                 .WORD   DTAERM
4253 017116                                  2$:       LET CKDFF := CKDFF + #1     ;INCREMENT # OF BYTES IN ERROR.
     017116  005237  017276                                                         INC     CKDFF
4254 017122  111437  003444                            MOVB (R4),TIME1             ;SAVE WAS DATA FOR TYPOUT.
4255 017126  042737  177400  003444                    BIC       #177400,TIME1         ;CLEAR GARBAGE.
4256 017134  111337  003446                            MOVB (R3),TIME2             ;SAVE SHOULD BE DATA FOR TYPOUT.
4257 017140  042737  177400  003446                    BIC       #177400,TIME2         ;CLEAR GARBAGE.
4258 017146                                            IF CKDFF LT #11. THEN       ;IF ERROR BYTE COUNT IS LESS THAN 11:
     017146  023727  017276  000013                                                 CMP     CKDFF,#11.
     017154  002017                                                                 BGE     50337$
4259 017156                                              PRINTX #DTAER2,R2,<B,TIME1>,<B,TIME2>  ;PRINT EXP + ACT DATA.
     017156  005046                                                                 CLR     -(SP)
     017160  153716  003446                                                         BISB    TIME2,(SP)
     017164  005046                                                                 CLR     -(SP)
     017166  153716  003444                                                         BISB    TIME1,(SP)
     017172  010246                                                                 MOV     R2,-(SP)
     017174  012746  004761                                                         MOV     #DTAER2,-(SP)
     017200  012746  000004                                                         MOV     #4,-(SP)
     017204  010600                                                                 MOV     SP,R0
     017206  104415                                                                 TRAP    C$PNTX
     017210  062706  000012                                                         ADD     #12,SP
4260 017214                                            ENDIF
     017214                                                                          50337$:
4261 017214  105723                          3$:       TSTB (R3)+                  ;UPDATE WRITE BUFFER ADDRESS.
4262 017216  105724                                    TSTB (R4)+                  ;UPDATE READ BUFFER ADDRESS.
```

```
4263 017220  105722                          TSTB (R2)+                          ;UPDATE BYTE COUNTER.
4264 017222                                  UNTIL R2 GT CKDCNT                  ;END OF DATA COMPARE REPEAT LOOP.
     017222  020237  017274                                                          CMP     R2,CKDCNT
     017226  003702                                                                  BLE     50333$
4265 017230                                  LET CKDCNT := CKDCNT + #1           ;CKDCNT EQUALS RECORD LENGTH.
     017230  005237  017274                                                          INC     CKDCNT
4266 017234                                  IF CKDFF NE #0 THEN                 ;IF COMPARE ERROR HAS OCCURED THEN:
     017234  005737  017276                                                          TST     CKDFF
     017240  001414                                                                  BEQ     50340$
4267 017242                                    PRINTB #DTAER3,CKDFF,CKDCNT       ;PRINT # OF BYTES IN ERROR.
     017242  013746  017274                                                          MOV     CKDCNT,-(SP)
     017246  013746  017276                                                          MOV     CKDFF,-(SP)
     017252  012746  005030                                                          MOV     #DTAER3,-(SP)
     017256  012746  000003                                                          MOV     #3,-(SP)
     017262  010600                                                                  MOV     SP,R0
     017264  104414                                                                  TRAP    C$PNTB
     017266  062706  000010                                                          ADD     #10,SP
4268 017272                                  ENDIF
     017272                                                                       50340$:
4269 017272                                ENDIF
     017272                                                                       50331$:
4270 017272                              ENDIF
     017272                                                                       50327$:
4271 017272  000207                        RTS     PC                            ;OTHERWISE, RETURN.
4272
4273 017274  000000              CKDCNT:   .WORD   0                             ;# OF BYTES TO BE VERIFIED -1.
4274 017276  000000              CKDFF:    .WORD   0                             ;# OF BYTES IN ERROR COUNTER.
```

```
4276                                    ;       SUBROUTINE TO FIND THE FIRST DEVICE IN THE TEST SEQUENCE.
4277                                    ;       INPUTS:
4278                                    ;       OUTPUTS:
4279                                    ;       REGISTERS:
4280                                    ;       CALLS:
4281
4282 017300                            FIRSTU::LET DROPED :B= #0                ;CLR UNIT DROPPED FLAG
     017300  105037  003527                                                             CLRB    DROPED
4283 017304                                    LET R5 := #0                     ;CLR DEVICE POINTER.
     017304  005005                                                                     CLR     R5
4284 017306                                    WHILE DEVTBL(R5) EQ #NINUSE DO   ;WHILE DEVICES ARE NOT IN USE:
     017306                                                                     50341$:
     017306  026527  002536  177774                                                     CMP     DEVTBL(R5),#NINUSE
     017314  001003                                                                     BNE     50342$
4285 017316                                      LET R5 := R5 + #2              ;POINT TO NEXT DEVICE.
     017316  062705  000002                                                             ADD     #2,R5
4286 017322                                    ENDDO
     017322  000771                                                                     BR      50341$
     017324                                                                     50342$:
4287 017324                                    IF DEVTBL(R5) EQ #END THEN       ;IF ALL UNITS HAVE BEEN DROPPED THEN:
     017324  026527  002536  177777                                                     CMP     DEVTBL(R5),#END
     017332  001001                                                                     BNE     50343$
4288 017334                                      DOCLN                            ;DO CLEAN CODE AND TERMINATE PASS.
     017334  104444                                                                     TRAP    C$DCLN
4289 017336                                    ENDIF
     017336                                                                     50343$:
4290 017336                                    LET L$LUN := DEVTBL(R5)          ;SET UNIT # IN "HEADER" FOR ERROR REPORT
     017336  016537  002536  002074                                                     MOV     DEVTBL(R5),L$LUN
4291 017344  000207                          RTS     PC                       ;RETURN WITH 1ST DEVICE IN R5.
4292
4293
4294
4295
4296
4297                                    ;       SUBROUTINE TO FIND THE NEXT UNIT IN THE TEST CYCLE.
4298                                    ;       INPUTS:
4299                                    ;       OUTPUTS:
4300                                    ;       REGISTERS:
4301                                    ;       CALLS:
4302
4303 017346                            NEXTU:: LET DROPED :B= #0                ;CLR UNIT DROPPED FLAG
     017346  105037  003527                                                             CLRB    DROPED
4304 017352  042705  177770                    BIC     #177770,R5
4305 017356                                    REPEAT                           ;REPEAT UNTIL THE NEXT DEVICE IS FOUND.
     017356                                                                     50344$:
4306 017356                                      LET R5 := R5 + #2              ;UPDATE DEVICE TABLE POINTER.
     017356  062705  000002                                                             ADD     #2,R5
4307 017362                                    UNTIL DEVTBL(R5) NE #NINUSE
     017362  026527  002536  177774                                                     CMP     DEVTBL(R5),#NINUSE
     017370  001772                                                                     BEQ     50344$
4308 017372                                    LET L$LUN := DEVTBL(R5)          ;SET UNIT # IN "HEADER" FOR ERROR REPORT
     017372  016537  002536  002074                                                     MOV     DEVTBL(R5),L$LUN
4309 017400  000207                          RTS     PC                       ;RETURN.
4310
4311
4312
```

```
4314                                    ;       SUBROUTINE TO DROP A DEVICE FROM THE TEST SEQUENCE.
4315                                    ;       INPUTS:
4316                                    ;       OUTPUTS:
4317                                    ;       REGISTERS:
4318                                    ;       CALLS:          MOVMSG, PRXST, LOG
4319
4320 017402                            DROPU:: LET R5SAVE := R5
     017402  010537  003460                                                  MOV     R5,R5SAVE
4321 017406                                    LET FTLCNT(R5) := FTLCNT(R5) + #1  ;INCREMENT THE FATAL ERROR COUNT.
     017406  005265  003320                                                  INC     FTLCNT(R5)
4322 017412                                    LET R4 := MSGPKT+MS.XS3 CLR.BY #377 ;GET UDIAG ERROR CODE FROM XSTAT3.
     017412  013704  002354                                                  MOV     MSGPKT+MS.XS3,R4
     017416  042704  000377                                                  BIC     #377,R4
4323 017422                                    LET R3 := MSGPKA(R5)            ;ADR OF THIS UNIT'S MSG PACKET.
     017422  016503  002506                                                  MOV     MSGPKA(R5),R3
4324 017426                                    LET R2 := #0                    ;CLR COUNTER.
     017426  005002                                                          CLR     R2
4325 017430                                    WHILE R2 NE #MSGCNT DO          ;WHILE THERE ARE MORE LOCATIONS:
     017430                                                                  50345$:
     017430  020227  000016                                                  CMP     R2,#MSGCNT
     017434  001405                                                          BEQ     50346$
4326 017436                                      LET (R3)+ := #-1              ;INIT THE MSG PACKET WITH ALL 1'S
     017436  012723  177777                                                  MOV     #-1,(R3)+
4327 017442                                      LET R2 := R2 + #2             ;UPDATE COUNTER.
     017442  062702  000002                                                  ADD     #2,R2
4328 017446                                    ENDDO
     017446  000770                                                          BR      50345$
     017450                                                                  50346$:
4329 017450                                    LET @TSDB(R5) := #GSCPK         ;INITIATE A GET STATUS COMMAND.
     017450  012775  002324  002456                                          MOV     #GSCPK,@TSDB(R5)
4330 017456  004737  011672                  JSR PC,WSSR                     ;WAIT A WHILE FOR SSR=1
4331 017462  004737  011726                  JSR PC,MOVMSG                   ;MOVE MSG PACKET TO COMMON AREA.
4332 017466                                    IF R4 EQ #X3.RNY THEN           ;IF WE HAVE A CAPSTAN RUNAWAY THEN:
     017466  020427  157400                                                  CMP     R4,#X3.RNY
     017472  001005                                                          BNE     50347$
4333 017474                                      ERRDF 16,RNYM,STAERM            ;REPORT CAPSTAN RUNAWAY WITH TACH CNT.
     017474  104455                                                          TRAP    C$ERDF
     017476  000020                                                          .WORD   16
     017500  004566                                                          .WORD   RNYM
     017502  005626                                                          .WORD   STAERM
4334 017504                                    ELSE                            ;ELSE-IF NOT A RUNNAWAY:
     017504  000402                                                          BR      50350$
     017506                                                                  50347$:
4335 017506  004737  017624                    JSR PC,PRXST                    ;PRINT EXTENDED STATUS REGISTERS.
4336 017512                                    ENDIF
     017512                                                                  50350$:
4337 017512                                    IFB RECLOG NE #0 THEN           ;IF THE RECORD HAS BEEN LOGGED THEN:
     017512  105737  003471                                                  TSTB    RECLOG
     017516  001404                                                          BEQ     50351$
4338 017520                                      LET DROPED :B= DROPED + #1    ;SET UNIT DROPPED FLAG.
     017520  105237  003527                                                  INCB    DROPED
4339 017524  004737  015436                    JSR PC,LOG                      ;LOG DATA BYTES + RD/WR ERRORS.
4340 017530                                    ENDIF
     017530                                                                  50351$:
4341 017530                                    DORPT                           ;PRINT PERFORMANCE REPORT
     017530  104424                                                          TRAP    C$DRPT
4342 017532                            DROPUA: IF PASCNT(R5) NE #0 THEN
```

```
        017532  005765  003260                                                  TST     PASCNT(R5)
        017536  001402                                                          BEQ     50352$
   4343 017540                            LET PASCNT(R5) := PASCNT(R5) - #1
        017540  005365  003260                                                  DEC     PASCNT(R5)
   4344 017544                            ENDIF
        017544                                                                  50352$:
   4345 017544                            LET DROPN := DEVTBL(R5)          ;SAVE # OF UNIT TO BE DROPPED.
        017544  016537  002536  017622                                          MOV     DEVTBL(R5),DROPN
   4346 017552                            LET R0 := R5 SHIFT -1            ;R0=LOGICAL DEVICE NUMBER
        017552  010500                                                          MOV     R5,R0
        017554  006200                                                          ASR     R0
   4347 017556                            DODU R0                  ;DROP THE UNIT: EXEC BGNDU-ENDDU CODE IF IDU = 0
        017556  104451                                                          TRAP    C$DODU
   4348 017560                            IF DEVTBL(R5) NE #NINUSE THEN    ;IF UNIT NOT DROPPED
        017560  026527  002536  177774                                          CMP     DEVTBL(R5),#NINUSE
        017566  001410                                                          BEQ     50353$
   4349 017570                              IFB IREC EQ #0 THEN              ;IF RECOVERY IS ENABLED THEN:
        017570  105737  002210                                                  TSTB    IREC
        017574  001005                                                          BNE     50354$
   4350 017576  000240                        NOP
   4351 017600  000240                        NOP
   4352 017602  000240                        NOP
   4353 017604                                LET STAFLG :B= STAFLG + #1      ;SET START FLAG TO ENABLE REWIND.
        017604  105237  003534                                                  INCB    STAFLG
   4354 017610                              ENDIF
        017610                                                                  50354$:
   4355 017610                            ENDIF
        017610                                                                  50353$:
   4356 017610                DRORTN: LET DROPED :B= DROPED + #1       ;SET UNIT DROPPED FLAG.
        017610  105237  003527                                                  INCB    DROPED
   4357 017614                            LET R5 := R5SAVE
        017614  013705  003460                                                  MOV     R5SAVE,R5
   4358 017620  000207                    RTS     PC                       ;RETURN.
   4359
   4360 017622  000000            DROPN:  .WORD   0                        ;# OF UNIT TO BE DROPPED
```

```
4362                                    ;       SUBROUTINE TO PRINT EXTENDED STATUS REGISTERS.
4363                                    ;       INPUTS:
4364                                    ;       OUTPUTS:
4365                                    ;       REGISTERS:
4366                                    ;       CALLS:
4367
4368 017624                            PRXST:: PRINTX #GETSTM
     017624  012746  005241                                                    MOV     #GETSTM,-(SP)
     017630  012746  000001                                                    MOV     #1,-(SP)
     017634  010600                                                            MOV     SP,R0
     017636  104415                                                            TRAP    C$PNTX
     017640  062706  000004                                                    ADD     #4,SP
4369 017644                                    PRINTX  #STAER5,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3
     017644  013746  002354                                                    MOV     MSGPKT+MS.XS3,-(SP)
     017650  013746  002352                                                    MOV     MSGPKT+MS.XS2,-(SP)
     017654  013746  002350                                                    MOV     MSGPKT+MS.XS1,-(SP)
     017660  013746  002346                                                    MOV     MSGPKT+MS.XS0,-(SP)
     017664  012746  006453                                                    MOV     #STAER5,-(SP)
     017670  012746  000005                                                    MOV     #5,-(SP)
     017674  010600                                                            MOV     SP,R0
     017676  104415                                                            TRAP    C$PNTX
     017700  062706  000014                                                    ADD     #14,SP
4370 017704  000207                                    RTS PC
4371
4372
4373
4374
4375                                    ;       SUBROUTINE TO HALT AFTER EACH COMMAND.
4376                                    ;       INPUTS:
4377                                    ;       OUTPUTS:
4378                                    ;       REGISTERS:      R3, R4
4379                                    ;       CALLS:
4380
4381 017706                            CKHAE:: IFB HAE NE #0 THEN              ;IF HALT FLAG IS SET:
     017706  105737  002206                                                    TSTB    HAE
     017712  001430                                                            BEQ     50355$
4382 017714                                      IFB MISCFG EQ #0 THEN          ;
     017714  105737  003537                                                    TSTB    MISCFG
     017720  001023                                                            BNE     50356$
4383 017722                                        MANUAL                      ;IS MANUAL INTERVENTION ALLOWED?
     017722  104450                                                            TRAP    C$MANI
4384 017724                                        BNCOMPLETE CKHRTN           ;BR IF NOT.
     017724  103023                                                            BCC     CKHRTN
4385 017726                                        LET R4 := CMDWRD            ;COMMAND WORD.
     017726  013704  003426                                                    MOV     CMDWRD,R4
4386 017732  004737  007654                       JSR PC,GCMDA                ;FETCH ADR OF CMD ASCII.
4387 017736                                        LET HALTM :B= (R3)+         ;MOVE CMD ASCII
     017736  112337  004124                                                    MOVB    (R3)+,HALTM
4388 017742                                        LET HALTM+1 :B= (R3)+
     017742  112337  004125                                                    MOVB    (R3)+,HALTM+1
4389 017746                                        LET HALTM+2 :B= (R3)                ;INTO MESSAGE.
     017746  111337  004126                                                    MOVB    (R3),HALTM+2
4390 017752                                        GMANIL HALTM,TIME1,1,YES    ;HALT - WAIT FOR AN OEPRATOR INPUT.
     017752  104443                                                            TRAP    C$GMAN
     017754  000404                                                            BR      10000$
     017756  003444                                                            .WORD   TIME1
     017760  000130                                                            .WORD   T$CODE
```

J10


```
        017762  004124                                                          .WORD   HALTM
        017764  000001                                                          .WORD   1
        017766                          10000$:
   4391 017766                                  ELSE
        017766  000402                                                  BR      50357$
        017770                                                          50356$:
   4392 017770                                    LET MISCFG :B= #0     ;
        017770  105037  003537                                          CLRB    MISCFG
   4393 017774                                  ENDIF
        017774                                                          50357$:
   4394 017774                                ENDIF
        017774                                                          50355$:
   4395 017774  000207                  CKHRTN: RTS     PC                      ;RETURN
```

```
4397                                    ;       SUBROUTINE TO CREATE THE SEQUENCE FOR A WRITE TAPE MARK
4398                                    ;       COMMAND.  WILL EXECUTE COMMAND TO UUT.
4399                                    ;       INPUTS:
4400                                    ;       OUTPUTS: CMDSEQ
4401                                    ;       CALLS:  SETUP, CMDAC, EXCUTE, GOWAIT
4402
4403 017776                            WRITEM::
4404 017776                                    LET R1 := #CMDSEQ
     017776  012701  003542                                                          MOV     #CMDSEQ,R1
4405 020002                                    LET (R1)+ := #WTM       ;COMMAND
     020002  012721  100011                                                          MOV     #WTM,(R1)+
4406 020006                                    LET (R1)+ := #1         ;BRF
     020006  012721  000001                                                          MOV     #1,(R1)+
4407 020012                                    LET (R1)+ := #1         ;ITERATIONS
     020012  012721  000001                                                          MOV     #1,(R1)+
4408 020016  005721                            TST (R1)+               ;PATTERN
4409 020020                                    LET (R1)+ := #END       ;TERMINATOR
     020020  012721  177777                                                          MOV     #END,(R1)+
4410 020024                                    LET R1 := #CMDSEQ       ;TOP OF BUFFER
     020024  012701  003542                                                          MOV     #CMDSEQ,R1
4411 020030  004737  007706                    JSR PC, SETUP           ;SET UP THE TABLE
4412 020034  004737  007600                    JSR PC, CMDAC           ;LOAD THE ASCII
4413 020040  004737  010654                    JSR PC, EXCUTE          ;ISSUE THE WTM COMMAND
4414 020044  004737  011274                    JSR PC, GOWAIT          ;WAIT FOR THE COMMAND TO FINISH
4415 020050  000207                            RTS PC                  ;RETURN TO CALLER
4416                                           .EVEN
4417
4418 020052                                    ENDMOD
```

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

020052                                  BGNMOD

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

020052                                  BGNRPT
020052                          L$RPT::

020052                                  LET     R5SAVE := R5            ;SAVE CURRENT DEVICE POINTER.
020052  010537  003460                                                      MOV     R5,R5SAVE
020056  004737  017300                  JSR     PC,FIRSTU               ;FIND THE FIRST UNIT.
020062                                  WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
020062                                                                          50360$:
020062  026527  002536  177777                                                  CMP     DEVTBL(R5),#END
020070  001562                                                                  BEQ     50361$
020072                                    PRINTS        #RPT1A,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
020072  016546  003330                                                          MOV     RECCNT(R5),-(SP)
020076  016546  003260                                                          MOV     PASCNT(R5),-(SP)
020102  016546  002536                                                          MOV     DEVTBL(R5),-(SP)
020106  012746  020714                                                          MOV     #RPT1A,-(SP)
020112  012746  000004                                                          MOV     #4,-(SP)
020116  010600                                                                  MOV     SP,R0
020120  104416                                                                  TRAP    C$PNTS
020122  062706  000012                                                          ADD     #12,SP
020126                                  PRINTS        #RPT1B,WRBC+30(R5),WRBC+20(R5),WRBC+10(R5),WRBC(R5)
020126  016546  002560                                                          MOV     WRBC(R5),-(SP)
020132  016546  002570                                                          MOV     WRBC+10(R5),-(SP)
020136  016546  002600                                                          MOV     WRBC+20(R5),-(SP)
020142  016546  002610                                                          MOV     WRBC+30(R5),-(SP)
020146  012746  020771                                                          MOV     #RPT1B,-(SP)
020152  012746  000005                                                          MOV     #5,-(SP)
020156  010600                                                                  MOV     SP,R0
020160  104416                                                                  TRAP    C$PNTS
020162  062706  000014                                                          ADD     #14,SP
020166                                  PRINTS        #RPT1C,RRBC+30(R5),RRBC+20(R5),RRBC+10(R5),RRBC(R5)
020166  016546  002620                                                          MOV     RRBC(R5),-(SP)
020172  016546  002630                                                          MOV     RRBC+10(R5),-(SP)
020176  016546  002640                                                          MOV     RRBC+20(R5),-(SP)
020202  016546  002650                                                          MOV     RRBC+30(R5),-(SP)
020206  012746  021042                                                          MOV     #RPT1C,-(SP)
020212  012746  000005                                                          MOV     #5,-(SP)
020216  010600                                                                  MOV     SP,R0
020220  104416                                                                  TRAP    C$PNTS
020222  062706  000014                                                          ADD     #14,SP
020226                                  PRINTS        #RPT1D,RFBC+30(R5),RFBC+20(R5),RFBC+10(R5),RFBC(R5)
020226  016546  002660                                                          MOV     RFBC(R5),-(SP)
020232  016546  002670                                                          MOV     RFBC+10(R5),-(SP)
020236  016546  002700                                                          MOV     RFBC+20(R5),-(SP)
020242  016546  002710                                                          MOV     RFBC+30(R5),-(SP)
020246  012746  021113                                                          MOV     #RPT1D,-(SP)
020252  012746  000005                                                          MOV     #5,-(SP)
```

```
        020256  010600                                                          MOV     SP,R0
        020260  104416                                                          TRAP    C$PNTS
        020262  062706  000014                                                  ADD     #14,SP
  4463  020266                  PRINTS        #RPT1F,WRREC(R5),RRREC(R5),RFREC(R5)
        020266  016546  002760                                                  MOV     RFREC(R5),-(SP)
        020272  016546  002740                                                  MOV     RRREC(R5),-(SP)
        020276  016546  002720                                                  MOV     WRREC(R5),-(SP)
        020302  012746  021217                                                  MOV     #RPT1F,-(SP)
        020306  012746  000004                                                  MOV     #4,-(SP)
        020312  010600                                                          MOV     SP,R0
        020314  104416                                                          TRAP    C$PNTS
        020316  062706  000012                                                  ADD     #12,SP
  4464  020322                  PRINTS        #RPT1G,WRUNR(R5),RRUNR(R5),RFUNR(R5)
        020322  016546  002770                                                  MOV     RFUNR(R5),-(SP)
        020326  016546  002750                                                  MOV     RRUNR(R5),-(SP)
        020332  016546  002730                                                  MOV     WRUNR(R5),-(SP)
        020336  012746  021270                                                  MOV     #RPT1G,-(SP)
        020342  012746  000004                                                  MOV     #4,-(SP)
        020346  010600                                                          MOV     SP,R0
        020350  104416                                                          TRAP    C$PNTS
        020352  062706  000012                                                  ADD     #12,SP
  4465  020356                  IFB BADTSW NE #0 THEN         ;
        020356  105737  002211                                          TSTB    BADTSW
        020362  001402                                                  BEQ     50362$
  4466  020364  004737  020446    JSR PC,BTRPT        ;GO PRINT BAD TAPE SPOTS WHEN  ENABLED
  4467  020370                  ENDIF
        020370                                                          50362$:
  4468  020370                  PRINTS        #RPT1I,SCCNT(R5),HRDCNT(R5),FTLCNT(R5),VFYCNT(R5)
        020370  016546  003300                                                  MOV     VFYCNT(R5),-(SP)
        020374  016546  003320                                                  MOV     FTLCNT(R5),-(SP)
        020400  016546  003310                                                  MOV     HRDCNT(R5),-(SP)
        020404  016546  003270                                                  MOV     SCCNT(R5),-(SP)
        020410  012746  021460                                                  MOV     #RPT1I,-(SP)
        020414  012746  000005                                                  MOV     #5,-(SP)
        020420  010600                                                          MOV     SP,R0
        020422  104416                                                          TRAP    C$PNTS
        020424  062706  000014                                                  ADD     #14,SP
  4469  020430  004737  017346      JSR   PC,NEXTU              ;FIND THE NEXT UNIT.
  4470  020434                  ENDDO
        020434  000612                                                  BR      50360$
        020436                                                          50361$:
  4471  020436                  LET R5 := R5SAVE              ;RESTORE CURRENT DEVICE POINTER.
        020436  013705  003460                                          MOV     R5SAVE,R5
  4472  020442                  EXIT    RPT
        020442  000167                                                  .WORD   J$JMP
        020444  001206                                                  .WORD   L10010-2-.
  4473
  4474                          ;       SUBR TO PRINT BAD TAPES SPOTS DURING THE REPORT PRINTS
  4475                          ;       WRITE RETRIES: CUMULATIVE COUNT
  4476                          ;       BAD TAPE SPOTS: COUNT PER TAPE PASS ONLY, NOT CUMULATIVE.
  4477                          ;       COUNT OF RECOVERABLE WRITE ERRORS EXCLUDES BAD TAPE SPOTS.
  4478
  4479  020446                  BTRPT:  PRINTS  #RPT1E,WRTYCT(R5)       ;PRINT GLOBAL WRITE RETRY COUNT
        020446  016546  003250                                                  MOV     WRTYCT(R5),-(SP)
        020452  012746  021341                                                  MOV     #RPT1E,-(SP)
        020456  012746  000002                                                  MOV     #2,-(SP)
        020462  010600                                                          MOV     SP,R0
```

```
                                                                                                TRAP    C$PNTS
        020464  104416                                                                          ADD     #6,SP
        020466  062706  000006
4480    020472                  LET BTPT := BTADDR(R5)  ;BTPT IS BOTH THE BAD TAPE SPOT COUNTER
        020472  016537  002550  003516                                                          MOV     BTADDR(R5),BTPT
4481    020500                  LET R3 := @BTPT SHIFT -1        ;AND THE LOGGING INDEX
        020500  017703  163012                                                                  MOV     @BTPT,R3
        020504  006203                                                                          ASR     R3
4482    020506                  PRINTS  #RPT1J,R3       ;PRINT # OF BAD TAPE SPOTS
        020506  010346                                                                          MOV     R3,-(SP)
        020510  012746  021371                                                                  MOV     #RPT1J,-(SP)
        020514  012746  000002                                                                  MOV     #2,-(SP)
        020520  010600                                                                          MOV     SP,R0
        020522  104416                                                                          TRAP    C$PNTS
        020524  062706  000006                                                                  ADD     #6,SP
4483    020530                  IF R3 NE #0 THEN                ;PRINT RECORD # IF BAD SPOTS DETECTED
        020530  005703                                                                          TST     R3
        020532  001457                                                                          BEQ     50363$
4484    020534                    IF R3 HI #20. THEN            ;
        020534  020327  000024                                                                  CMP     R3,#20.
        020540  101402                                                                          BLOS    50364$
4485    020542                      LET R3 := #20.              ;20 BAD SPOTS IS THE LIMIT
        020542  012703  000024                                                                  MOV     #20.,R3
4485    020546                    ENDIF
        020546                                                                                  50364$:
4487    020546                    PRINTS        #CRLFSP         ;
        020546  012746  005300                                                                  MOV     #CRLFSP,-(SP)
        020552  012746  000001                                                                  MOV     #1,-(SP)
        020556  010600                                                                          MOV     SP,R0
        020560  104416                                                                          TRAP    C$PNTS
        020562  062706  000004                                                                  ADD     #4,SP
4488    020566                  LET R4 := BTPT + #2     ;FETCH A BAD SPOT ID
        020566  013704  003516                                                          MOV     BTPT,R4
        020572  062704  000002                                                          ADD     #2,R4
4489    020576                  LET R2 := #0            ;R2 = PRINT COUNT PER LINE: 10 MAX
        020576  005002                                                                  CLR     R2
4490    020600                  REPEAT                          ;
        020600                                                                          50365$:
4491    020600                    PRINTS      #RPT1K,(R4)       ;PRINT A BAD SPOT ID
        020600  011446                                                                          MOV     (R4),-(SP)
        020602  012746  021451                                                                  MOV     #RPT1K,-(SP)
        020606  012746  000002                                                                  MOV     #2,-(SP)
        020612  010600                                                                          MOV     SP,R0
        020614  104416                                                                          TRAP    C$PNTS
        020616  062706  000006                                                                  ADD     #6,SP
4492    020622                    LET R2 := R2 + #1     ;COUNT PRINTS
        020622  005202                                                                  INC     R2
4493    020624                    LET R4 := R4 + #2     ;NEXT
        020624  062704  000002                                                          ADD     #2,R4
4494    020630                    IF R2 EQ #10. THEN    ;
        020630  020227  000012                                                          CMP     R2,#10.
        020634  001014                                                                  BNE     50366$
4495    020636                      PRINTS #CRLFSP              ;GO TO NEXT PRINT LINE PAST 10 PRINTS
        020636  012746  005300                                                                  MOV     #CRLFSP,-(SP)
        020642  012746  000001                                                                  MOV     #1,-(SP)
        020646  010600                                                                          MOV     SP,R0
        020650  104416                                                                          TRAP    C$PNTS
        020652  062706  000004                                                                  ADD     #4,SP
```

```
4496 020656                                    LET R3 := R3 - #10.          ;ADJUST BAD SPOT COUNT
     020656  162703  000012                                                          SUB     #10.,R3
4497 020662                                    LET R2 := R2 - #10.          ;ADJUST PRINT COUNT
     020662  162702  000012                                                          SUB     #10.,R2
4498 020666                                  ENDIF                          ;
     020666                                                                         50366$:
4499 020666                                UNTIL R2 EQ R3                   ;LIMIT: # OF BAD SPOTS
     020666  020203                                                                 CMP     R2,R3
     020670  001343                                                                 BNE     50365$
4500 020672                                ENDIF                            ;
     020672                                                                         50363$:
4501 020672                                PRINTS  #CRLF                    ;
     020672  012746  005275                                                                 MOV     #CRLF,-(SP)
     020676  012746  000001                                                                 MOV     #1,-(SP)
     020702  010600                                                                 MOV     SP,R0
     020704  104416                                                                 TRAP    C$PNTS
     020706  062706  000004                                                                 ADD     #4,SP
4502 020712  000207                        RTS PC
4503
4515                                       .NLIST  BEX
4516 020714      045      116      045  RPT1A: .ASCIZ  /<N><N>AUNIT <D1><S3>APASS:<D5><S3>ARECORD:<D5><N>/
4517 020771      045      101      102  RPT1B: .ASCIZ  /<A>BYTES WRITTEN  <D3><A>,<Z3><A>,<Z3><A>,<Z3><N>/
4518 021042      045      101      102  RPT1C: .ASCIZ  /<A>BYTES READ REV <D3><A>,<Z3><A>,<Z3><A>,<Z3><N>/
4519 021113      045      101      102  RPT1D: .ASCII  /<A>BYTES READ FWD <D3><A>,<Z3><A>,<Z3><A>,<Z3><N>/
4520 021163      045      123      062        .ASCIZ  /<S23><AWRT><S4><ARDR><S4><ARDF><N>/
4521 021217      045      101      122  RPT1F: .ASCIZ  /<A>RECOVERABLE ERRORS   <D5><S2><D5><S2><D5><N>/
4522 021270      045      101      125  RPT1G: .ASCIZ  /<A>UNRECOVERABLE ERRORS <D5><S2><D5><S2><D5><N>/
4523 021341      045      101      127  RPT1E: .ASCIZ  /<A>WRITE RETRIES<S8><D5><N>/
4524 021371      045      116      045  RPT1J: .ASCIZ  /<N><D2><A BAD SPOTS THIS PASS PRECEDING RECORD #:/
4525 021451      045      104      065  RPT1K: .ASCIZ  /<A><D5><S1>/
4526 021460      045      101      123  RPT1I: .ASCII  "<A>SPEC COND<S3><AHARD><S3><AFATAL><S3><ACOMPARE><N"
4527 021534      045      123      063        .ASCIZ  /<S3><D5><S3><D5><S3><D5><S3><D5><N><N>/
4528 021571      045      116      045  TAPCAP: .ASCIZ  /<N><N>ATAPE IN CARTRIDGE MUST BE 600' IN LENGTH.<N><N>/
4529                                       .LIST   BEX
4530                                       .EVEN
4531 021654                                ENDRPT
     021654                            L10010:
     021654  104425                                                                 TRAP    C$RPT
4532
4533                                   ;LOAD DEVICE PROTECTION TABLE
4534                                   ;TABLE FOR SUPERVISOR TO IDENTIFY THE P TBL FOR THE LOAD DEV
4535                                   ;THE SUPERVISOR USES THE TBL TO WARN THE OPERATOR WHEN HE TRIES TO TEST THE LOAD DEV
4536
4537 021656                                BGNPROT
     021656                            L$PROT::
4538 021656  000000                          .WORD 0                  ;P-TBL OFFSET OF TSSR, THE TK25 CSR
4539 021660  177777                          .WORD -1                 ;P-TBL OFFSET OF MASS BUS UNIT #: -1 = NOT A MASS BUS DEV
4540 021662  177777                          .WORD -1                 ;P-TBL OFFSET OF DRIVE #: -1 = NONE, ONE DRIVE PER UNIBUS AD
4541 021664                                ENDPROT
```

```
                                .SBTTL  INITIALIZE SECTION

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

                                        BGNINIT
                                L$INIT::

                                INIT10: IF @BIT0!BIT1 SETIN @CMDPKT THEN  ;IF CMD PACKET IS NOT ON MODULO 4 BOUNDRY:
021664  032727  000003  002314                                              BIT     @BIT0!BIT1,@CMDPKT
021672  001421                                                              BEQ     50367$
                                          ERRSF 1,CMDPKM                ;PRINT ERROR MSG,
021674  104454                                                            TRAP    C$ERSF
021676  000001                                                            .WORD   1
021700  004164                                                            .WORD   CMDPKM
021702  000000                                                            .WORD   0
                                          DELAY 20.                     ;GO TO SUPERVISOR, WAIT 2 SECONDS.
021704  012727  000024                                                    MOV     @20.,(PC)+
021710  000000                                                            .WORD   0
021712  013727  002116                                                    MOV     L$DLY,(PC)+
021716  000000                                                            .WORD   0
021720  005367  177772                                                    DEC     -6(PC)
021724  001375                                                            BNE     .-4
021726  005367  177756                                                    DEC     -22(PC)
021732  001367                                                            BNE     .-20
                                        BR      INIT10                  ;
                                      ENDIF                             50367$:

                                IFB CLRFLG NE @0 THEN           ;IF CLR COUNTERS FLAG SET:
021736  105737  002204                                                    TSTB    CLRFLG
021742  001413                                                            BEQ     50370$
                                  CLRB  CLRFLG                  ;INIT CLR FLAG.
021744  105037  002204                                                    CLR     R2
                                  LET R2 := @0

                                  WHILE R2 NE @CNTLEN DO        50371$:
021750  005002                                                            CMP     R2,@CNTLEN
021752  020227  000550                                                    BEQ     50372$
021756  001405
                                    LET WRBC(R2) := @0          ;CLR ALL STATISTICAL COUNTERS.
021760  005062  002560                                                    CLR     WRBC(R2)
                                    LET R2 := R2 + @2
021764  062702  000002                                                    ADD     @2,R2
                                  ENDDO
021770  000770                                                            BR      50371$
                                                                          50372$:
                                ENDIF                           50370$:

                                IFB RRANV NE @0 THEN            ;IF RESET RANDOM VARIABLE FLAG IS SET THEN:
021772  105737  002205                                                    TSTB    RRANV
021776  001406                                                            BEQ     50373$
                                  LET RANB := @RANBC            ;RESET RANDOM BASE @.
022000  012737  153624  003440                                            MOV     @RANBC,RANB
                                  LET RANS := @RANSC            ;RESET RANDOM SAVE LOCATION.
```

D11


```
022006  012737  032561  003442                                            MOV     #RANSC,RANS
022014                        ENDIF
022014                                                                    50373$:
022014                        READEF  #EF.START           ;READ START COMMAND EVENT FLAG.
022014  012700  000040                                                            MOV     #EF.START,R0
022020  104447                                                                    TRAP    C$REFG
022022                        BNCOMPLETE      INIT15      ;BRANCH IF NOT STARTING.
022022  103030                                                                    BCC     INIT15
022024                        LET STAFLG :B= STAFLG + #1  ;SET START COMMAND FLAG.
022024  105237  003534                                            INCB    STAFLG
022030                        LET HERE :B= #0             ;RESET THE TEST 3 ASCII SEMAPHORE
022030  105037  003340                                            CLRB    HERE
022034                        LET R5 := #6
022034  012705  000006                                            MOV     #6,R5
022040                        REPEAT                      ;INITIATE UNIT NUMBER TABLE
022040                                                    50374$:
022040                          LET DEVTBL(R5) := #NINUSE ;BY STORING NOT IN USE IN EACH LOCATION.
022040  012765  177774  002536                                    MOV     #NINUSE,DEVTBL(R5)
022046                          LET R5 := R5 - #2
022046  162705  000002                                            SUB     #2,R5
022052                        UNTIL R5 EQ #0
022052  005705                                                    TST     R5
022054  001371                                                    BNE     50374$
022056                        LET R5 := L$UNIT SHIFT 1
022056  013705  002012                                            MOV     L$UNIT,R5
022062  006305                                                    ASL     R5
022064                        REPEAT                      ;STORE ALL UNIT
022064                                                    50375$:
022064                          LET R5 := R5 - #2         ;NUMBERS IN DEVTBL.
022064  162705  000002                                            SUB     #2,R5
022070                          LET DEVTBL(R5) := R5 SHIFT -1
022070  010565  002536                                            MOV     R5,DEVTBL(R5)
022074  006265  002536                                            ASR     DEVTBL(R5)
022100                        UNTIL R5 EQ #0
022100  005705                                                    TST     R5
022102  001370                                                    BNE     50375$

022104                INIT15: READEF #EF.PWR              ;HAS THERE BE A POWER FAILURE?
022104  012700  000034                                                    MOV     #EF.PWR,R0
022110  104447                                                            TRAP    C$REFG
022112                        BNCOMPLETE INIT16           ;BRANCH IF NOT.
022112  103004                                                            BCC     INIT16
022114                          LET STAFLG :B= STAFLG + #1  ;IF SO - SET THE START FLAG.
022114  105237  003534                                            INCB    STAFLG
022120                          LET PWRFLG :B= PWRFLG + #1 ;IF SO - SET THE POWER FAIL FLAG.
022120  105237  003535                                            INCB    PWRFLG

022124                INIT16: RFLAGS  OPFLAG              ;READ AND STORE FLAGS SET BY OPERATOR
022124  104421                                                            TRAP    C$RFLA
022126  010037  003540                                                    MOV     R0,OPFLAG
022132                        LET R3 := #0                ;CLEAR EVENT FLAG
022132  005003                                                    CLR     R3
022134                            IFB PWRFLG EQ #0 THEN   ;IF POWER FAIL HAS NOT OCCURRED THEN:
022134  105737  003535                                            TSTB    PWRFLG
022140  001020                                                    BNE     50376$
022142                              READEF #EF.NEW        ;UPDATE PASS COUNT WHEN
022142  012700  000035                                                    MOV     #EF.NEW,R0
```

```
     022146 104447                                                                 TRAP    C$REFG
4604 022150                                  IFCOND CS THEN          ;SUPERVISOR IS IN NEW PASS
     022150 103014                                                         BCC     50377$
4605 022152                                    IFB STAFLG EQ #0 THEN  ;AND DIAG WAS NEITHER STARTED
     022152 105737 003534                                                  TSTB    STAFLG
     022156 001010                                                         BNE     50400$
4606 022160                                      READEF #EF.RES        ;NOR
     022160 012700 000037                                                          MOV     #EF.RES,RO
     022164 104447                                                                 TRAP    C$REFG
4607 022166                                      IFCOND CC THEN        ;RESTARTED
     022166 103402                                                         BCS     50401$
4608 022170                                        LET R3 := COMP R3   ;DO IT
     022170 005103                                                         COM     R3
4609 022172                                      ELSE
     022172 000401                                                         BR      50402$
     022174                                                                50401$:
4610 022174                                        LET R3 := R3 + #1   ;SET 1ST PASS IF NEW PASS AND
     022174 005203                                                         INC     R3
4611 022176                                      ENDIF                 ;RESTARTING
     022176                                                                50402$:
4612 022176                                    ELSE
     022176 000401                                                         BR      50403$
     022200                                                                50400$:
4613 022200                                      LET R3 := R3 + #1     ;SET 1ST PASS IF NEW PASS AND
     022200 005203                                                         INC     R3
4614 022202                                    ENDIF                   ;STARTING
     022202                                                                50403$:
4615 022202                                  ENDIF                     ;DO NOT UPDATE IT ON CONTINUE
     022202                                                                50377$:
4616 022202                                ENDIF                       ;OR ON POWER FAIL
     022202                                                                50376$:
4617 022202 004737 017300            JSR     PC,FIRSTU                 ;INIT DEVICE POINTER.
4618 022206                          LET R2 := #0                      ;INIT DEVICE COUNTER.
     022206 005002                                                         CLR     R2
4619 022210                          WHILE DEVTBL(R5) NE #END DO
     022210                                                                50404$:
     022210 026527 002536 177777                                           CMP     DEVTBL(R5),#END
     022216 001450                                                         BEQ     50405$
4620 022220                            LET R2 := R2 + #1
     022220 005202                                                         INC     R2
4621 022222                            LET RO := R5 SHIFT -1
     022222 010500                                                         MOV     R5,RO
     022224 006200                                                         ASR     RO
4622 022226                            GPHARD  RO,RO                   ;GET HARDWARE P TABLE FROM SUPER.
     022226 104442                                                                 TRAP    C$GPHRD
4623 022230                            IFCOND CS THEN
     022230 103036                                                         BCC     50406$
4624 022232                              LET TSSR(R5) := (RO)          ;SAVE TSSR ADDRESS.
     022232 011065 002466                                                  MOV     (RO),TSSR(R5)
4625 022236                              LET TSDB(R5) := (RO)+ - #2    ;SAVE TSDB ADDRESS.
     022236 012065 002456                                                  MOV     (RO)+,TSDB(R5)
     022242 162765 000002 002456                                           SUB     #2,TSDB(R5)
4626 022250                              LET TSVCT(R5) := (RO)         ;SAVE INTERRUPT VECTOR ADDRESS.
     022250 011065 002476                                                  MOV     (RO),TSVCT(R5)
4627 022254                              SETVEC TSVCT(R5),TS4INT(R5),#INTPRI ;SET UP INTERUPT PROCESSING CONDITIONS.
     022254 012746 000340                                                          MOV     #INTPRI,-(SP)
     022260 016546 002516                                                          MOV     TS4INT(R5),-(SP)
```

```
        022264  016546  002476                                                  MOV     TSVCT(R5),-(SP)
        022270  012746  000003                                                  MOV     #3,-(SP)
        022274  104437                                                          TRAP    C$SVEC
        022276  062706  000010                                                  ADD     #10,SP
  4628  022302                    IF R3 NE #0 THEN            ;ACTUAL PASSCOUNT UPDATE PER R3
        022302  005703                                                         TST     R3
        022304  001410                                                         BEQ     50407$
  4629  022306                      IF R3 LT #0 THEN
        022306  005703                                                         TST     R3
        022310  002003                                                         BGE     50410$
  4630  022312                        LET PASCNT(R5) := PASCNT(R5) + #1
        022312  005265  003260                                                 INC     PASCNT(R5)
  4631  022316                      ELSE
        022316  000403                                                         BR      50411$
        022320                                                                50410$:
  4632  022320                        LET PASCNT(R5) := #1
        022320  012765  000001  003260                                         MOV     #1,PASCNT(R5)
  4633  022326                      ENDIF
        022326                                                                50411$:
  4634  022326                    ENDIF
        022326                                                                50407$:
  4635  022326                  ENDIF
        022326                                                                50406$:
  4636  022326                  LET RECCNT(R5) := #0          ;CLEAR RECORD COUNT
        022326  005065  003330                                                 CLR     RECCNT(R5)
  4637  022332  004737  017346               JSR   PC,NEXTU                 ;DO IT FOR ALL DEVICES.
  4638  022336                ENDDO
        022336  000724                                                         BR      50404$
        022340                                                                50405$:
  4639
  4640  022340                IF R2 EQ #0 THEN              ;IF THERE ARE NO UNITS:
        022340  005702                                                         TST     R2
        022342  001026                                                         BNE     50412$
  4641  022344                  PRINTF #AUDRPM              ;PRINT ALL UNITS DROPPED.
        022344  012746  004727                                                 MOV     #AUDRPM,-(SP)
        022350  012746  000001                                                 MOV     #1,-(SP)
        022354  010600                                                         MOV     SP,R0
        022356  104417                                                         TRAP    C$PNTF
        022360  062706  000004                                                 ADD     #4,SP
  4642  022364                  DELAY 20.                   ;GO TO SUPERVISOR, WAIT 2 SECONDS.
        022364  012727  000024                                                 MOV     #20.,(PC)+
        022370  000000                                                         .WORD   0
        022372  013727  002116                                                 MOV     L$DLY,(PC)+
        022376  000000                                                         .WORD   0
        022400  005367  177772                                                 DEC     -6(PC)
        022404  001375                                                         BNE     .-4
        022406  005367  177756                                                 DEC     -22(PC)
        022412  001367                                                         BNE     .-20
  4643  022414                  BREAK                       ;GO TO SUPERVISOR, CHECK TTY.
        022414  104422                                                         TRAP    C$BRK
  4644  022416                DOCLN                         ;DO CLEAN CODE + ABORT PASS.
        022416  104444                                                         TRAP    C$DCLN
  4645  022420                ENDIF
        022420                                                                50412$:
  4646
  4647  022420                SETPRI #PRI00                 ;LOWER CPU PRIORITY TO 0
        022420  012700  000000                                                 MOV     #PRI00,R0
```

```
      022424  104441                                                    TRAP    C$SPRI
4648  022426                  IFB IREC EQ #0 AND #ADR NOTSETIN OPFLAG THEN    ;IF ERROR RECOVERY IS ENABLED
      022426  105737  002210                                            TSTB    IREC
      022432  001145                                                    BNE     50413$
      022434  032737  000020  003540                                    BIT     #ADR,OPFLAG
      022442  001141                                                    BNE     50413$
4649  022444  004737  017300            JSR PC,FIRSTU                 ;AND AUTO-DROP NOT CALLED, THEN SET UP FOR FIRST UNI
4650  022450                            WHILE DEVTBL(R5) NE #END DO   ;WHILE THERE ARE MORE DEVICES:
      022450                                                          50414$:
      022450  026527  002536  177777                                    CMP     DEVTBL(R5),#END
      022456  001533                                                    BEQ     50415$
4651  022460                              BEGIN COUNTER               ;START 3.5 MINUTE COUNTER
4652  022460                                INCR TIME1 FROM #1 TO #25 BY #1
      022460  012737  000001  003444                                    MOV     #1,TIME1
      022466  000402                                                    BR      50417$
      022470                                                          50420$:
      022470  005237  003444                                          INC     TIME1
      022474                                                          50417$:
      022474  023727  003444  000025                                    CMP     TIME1,#25
      022502  003106                                                    BGT     50421$
4653  022504                                LET @TSDB(R5) := #GSCPK   ;AND GET UNITS STATUS
      022504  012775  002324  002456                                    MOV     #GSCPK,@TSDB(R5)
4654  022512                                DELAY 250.                ;WAIT 25 MSEC.
      022512  012727  000372                                                    MOV     #250.,(PC)+
      022516  000000                                                    .WORD   0
      022520  013727  002116                                                    MOV     L$DLY,(PC)+
      022524  000000                                                    .WORD   0
      022526  005367  177772                                                    DEC     -6(PC)
      022532  001375                                                            BNE     .-4
      022534  005367  177756                                                    DEC     -22(PC)
      022540  001367                                                            BNE     .-20
4655  022542                  IF #TS.SSR SETIN @TSSR(R5) THEN
      022542  032775  000200  002466                                    BIT     #TS.SSR,@TSSR(R5)
      022550  001420                                                    BEQ     50422$
4656  022552                    IF #TS.OFL NOTSETIN @TSSR(R5) THEN
      022552  032775  000100  002466                                    BIT     #TS.OFL,@TSSR(R5)
      022560  001001                                                    BNE     50423$
4657  022562                      LEAVE COUNTER               ;EXIT COUNTER WHEN UNIT ON LINE
      022562  000456                                                    BR      50416$
4658  022564                    ELSE
      022564                                                          50423$:
4659  022564                      PRINTF #OFLINM,DEVTBL(R5) ;PRINT UNIT OFF LINE EVERY 10 SEC
      022564  016546  002536                                                    MOV     DEVTBL(R5),-(SP)
      022570  012746  005211                                                    MOV     #OFLINM,-(SP)
      022574  012746  000002                                                    MOV     #2,-(SP)
      022600  010600                                                            MOV     SP,R0
      022602  104417                                                            TRAP    C$PNTF
      022604  062706  000006                                                    ADD     #6,SP
4660  022610                    ENDIF
      022610                                                          50424$:
4661  022610                  ELSE
      022610  000412                                                    BR      50425$
      022612                                                          50422$:
4662  022612                    PRINTF #NRDYM,DEVTBL(R5)
      022612  016546  002536                                                    MOV     DEVTBL(R5),-(SP)
      022616  012746  023544                                                    MOV     #NRDYM,-(SP)
      022622  012746  000002                                                    MOV     #2,-(SP)
```

```
        022626  010600                                                          MOV     SP,R0
        022630  104417                                                          TRAP    C$PNTF
        022632  062706  000006                                                  ADD     #6,SP
   4663 022636                          ENDIF
        022636                                                          50425$:
   4664 022636                          INCR TIME2 FROM #1 TO #13 BY #1
        022636  012737  000001  003446                                          MOV     #1,TIME2
        022644  000402                                                         BR      50426$
        022646                                                          50427$:
        022646  005237  003446                                                  INC     TIME2
        022652                                                          50426$:
        022652  023727  003446  000013                                          CMP     TIME2,#13
        022660  003016                                                         BGT     50430$
   4665 022662                            DELAY 200.          ;WAIT FOR UNIT TO BE SET ON-LINE
        022662  012727  000310                                                  MOV     #200.,(PC)+
        022666  000000                                                         .WORD   0
        022670  013727  002116                                                  MOV     L$DLY,(PC)+
        022674  000000                                                         .WORD   0
        022676  005367  177772                                                  DEC     -6(PC)
        022702  001375                                                         BNE     .-4
        022704  005367  177756                                                  DEC     -22(PC)
        022710  001367                                                         BNE     .-20
   4666 022712                            BREAK               ;ALLOW TERMINAL INTERRUPT
        022712  104422                                                         TRAP    C$BRK
   4667 022714                          ENDINC
        022714  000754                                                         BR      50427$
        022716                                                          50430$:
   4668 022716                        ENDINC
        022716  000664                                                         BR      50420$
        022720                                                          50421$:
   4669 022720                            END COUNTER
        022720                                                          50416$:
   4670 022720                            IF TIME1 GT #25 THEN        ;IF OFF LINE FOR 3.5 MINUTES
        022720  023727  003444  000025                                          CMP     TIME1,#25
        022726  003404                                                         BLE     50431$
   4671 022730  004737  01172o              JSR PC,MOVMSG           ;GET MESSAGE PACKET
   4672 022734  004737  012762              JSR PC,TCC1             ;PRINT ERROR AND DROP OFF LINE UNIT
   4673 022740                            ENDIF
        022740                                                          50431$:
   4674                                   ;REPEAT UNTIL ON LINE OR TIMED OUT.
   4675 022740  004737  017346            JSR PC,NEXTU            ;SET UP FOR NEXT UNIT.
   4676 022744                          ENDDO
        022744  000641                                                         BR      50414$
        022746                                                          50415$:
   4677 022746                        ENDIF
        022746                                                          50413$:
   4678 022746                        IFB PWRFLG EQ #0 THEN
        022746  105737  003535                                                  TSTB    PWRFLG
        022752  001026                                                         BNE     50432$
   4679 022754                            MEMORY DATAWT               ;REQUEST MEMORY FROM SUPER FOR RD/WR BUFFERS.
        022754  104431                                                         TRAP    C$MEM
        022756  010037  003414                                                  MOV     R0,DATAWT
   4680 022762                            LET DATARD := DATAWT + #DATCNT      ;SET RD BFR AD
        022762  013737  003414  003416                                          MOV     DATAWT,DATARD
        022770  062737  010000  003416                                          ADD     #DATCNT,DATARD
   4681 022776                            IF @DATAWT LT #DATCNT THEN  ;WHEN NOT ENOUGH FREE MEMO AVAILABLE
        022776  027727  160412  010000                                          CMP     @DATAWT,#DATCNT
```

```
        023004  002011                                                          BGE     50433$
4682    023006                          PRINTF #MEMOM                   ;WARN OPERATOR
        023006  012746  023054                                                  MOV     #MEMOM,-(SP)
        023012  012746  000001                                                  MOV     #1,-(SP)
        023016  010600                                                          MOV     SP,R0
        023020  104417                                                          TRAP    C$PNTF
        023022  062706  000004                                                  ADD     #4,SP
4683    023026                          DOCLN                           ;AND ABORT PASS
        023026  104444                                                          TRAP    C$DCLN
4684    023030                        ENDIF           ;DIAG MUST BE RE-LOADED IN A CPU WITH LARGER MEMO
        023030                                                          50433$:
4685    023030                      ENDIF
        023030                                                          50432$:
4686
4687    023030                      LET CHGFLG :B= #0                   ;CLR CHANGE CMD SEQ TBL FLAG.
        023030  105037  002217                                                  CLRB    CHGFLG
4688    023034                      LET R3 := #ENDFLG
        023034  012703  003534                                                  MOV     #ENDFLG,R3
4689    023040  004737  011656      JSR     PC,CLRERR               ;CLEAR ALL FLAGS.
4690    023044                      LET PWRFLG :B= #0               ;CLEAR THE POWER FAIL FLAG.
        023044  105037  003535                                                  CLRB    PWRFLG
4691
4692    023050                      EXIT    INIT
        023050  104432                                                          TRAP    C$EXIT
        023052  000104                                                          .WORD   L10012-.
4704    023054     045     101     106  MEMOM:  .ASCII  /%AFREE MEMO TOO SMALL FOR RD-WR BFRS%N/
        023057     122     105     105
        023062     040     115     105
        023065     115     117     040
        023070     124     117     117
        023073     040     123     115
        023076     101     114     114
        023101     040     106     117
        023104     122     040     122
        023107     104     055     127
        023112     122     040     102
        023115     106     122     123
        023120     045     116
4705    023122     045     101     122         .ASCIZ  /%ARE-LOAD IN LARGER MEMO%N/
        023125     105     055     114
        023130     117     101     104
        023133     040     111     116
        023136     040     114     101
        023141     122     107     105
        023144     122     040     115
        023147     105     115     117
        023152     045     116     000
4706                                    .EVEN
4707
4708    023156                      ENDINIT
        023156                  L10012:
        023156  104411                                                          TRAP    C$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
                                ;--

023160                                  BGNAUTO
023160                          L$AUTO::

                                        JSR PC,FIRSTU                   ;FIND FIRST UNIT
                                        WHILE DEVTBL(R5) NE #END DO     ;
023164                                                                          50434$:
023164  026527  002536  177777                                                  CMP     DEVTBL(R5),#END
023172  001525                                                                  BEQ     50435$
                                          LET TRAPD4 :B= #0             ;
023174  105037  003536                                                          CLRB    TRAPD4
                                        SETVEC #4,#TRAP4,#PRIO7         ;SET VECTOR 4
023200  012746  000340                                                          MOV     #PRIO7,-(SP)
023204  012746  023574                                                          MOV     #TRAP4,-(SP)
023210  012746  000004                                                          MOV     #4,-(SP)
023214  012746  000003                                                          MOV     #3,-(SP)
023220  104437                                                                  TRAP    C$SVEC
023222  062706  000010                                                          ADD     #10,SP
                                        LET R2 := @TSSR(R5)             ;ADDRESS TK25 INTERFACE
023226  017502  002466                                                          MOV     @TSSR(R5),R2
                                        CLRVEC #4                       ;CLEAR VECTOR AT 4
023232  012700  000004                                                          MOV     #4,R0
023236  104436                                                                  TRAP    C$CVEC
                                        IFB TRAPD4 NE #0 THEN
023240  105737  003536                                                          TSTB    TRAPD4
023244  001423                                                                  BEQ     50436$
                                          LET FTLCNT(R5) := FTLCNT(R5) + #1
023246  005265  003320                                                          INC     FTLCNT(R5)
                                          PRINTF #AUTODM,TSSR(R5)       ;PRINT ERROR
023252  016546  002466                                                          MOV     TSSR(R5),-(SP)
023256  012746  023450                                                          MOV     #AUTODM,-(SP)
023262  012746  000002                                                          MOV     #2,-(SP)
023266  010600                                                                  MOV     SP,R0
023270  104417                                                                  TRAP    C$PNTF
023272  062706  000006                                                          ADD     #6,SP
                                          LET DROPN := DEVTBL(R5)               ;SAVE # OF UNIT TO BE DROPPED
023276  016537  002536  017622                                                  MOV     DEVTBL(R5),DROPN
                                          LET R0 := R5 SHIFT -1                 ;R0=LOGICAL DEVICE NUMBER
023304  010500                                                                  MOV     R5,R0
023306  006200                                                                  ASR     R0
                                          DODU R0                       ;DROP THE UNIT: EXEC BGNDU-ENDDU CODE IF IDU = 0
023310  104451                                                                  TRAP    C$DODU
                                        ELSE
023312  000452                                                                  BR      50437$
023314                                                                          50436$:
                                          LET @TSDB(R5) := #GSCPK       ;SEND GET STATUS COMMAND
023314  012775  002324  002456                                                  MOV     #GSCPK,@TSDB(R5)
                                        JSR PC,WSSR                     ;WAIT
                                        IF #TS.SSR SETIN @TSSR(R5) THEN
023326  032775  000200  002466                                                  BIT     #TS.SSR,@TSSR(R5)
```

```
      023334  001423
 4736 023336
      023336  032775  000100  002466          IF #TS.OFL SETIN @TSSR(R5) THEN
      023344  001416
 4737 023346
      023346  005265  003320                    LET FTLCNT(R5) := FTLCNT(R5) + #1
 4738 023352
      023352  016546  002536                    PRINTF #OFLINM,DEVTBL(R5)
      023356  012746  005211
      023362  012746  000002
      023366  010600
      023370  104417
      023372  062706  000006
 4739 023376  004737  017532                    JSR PC,DROPUA
 4740 023402                                   ENDIF
      023402
 4741 023402                                 ELSE
      023402  000416
      023404
 4742 023404                                   LET FTLCNT(R5) := FTLCNT(R5) + #1
      023404  005265  003320
 4743 023410                                   PRINTF #NRDYM,DEVTBL(R5)
      023410  016546  002536
      023414  012746  023544
      023420  012746  000002
      023424  010600
      023426  104417
      023430  062706  000006
 4744 023434  004737  017532                    JSR PC,DROPUA
 4745 023440                                   ENDIF
      023440
 4746 023440                                   ENDIF
      023440
 4747 023440  004737  017346                    JSR PC,NEXTU
 4748 023444                                   ENDDO
      023444  000647                           BR      50434$
      023446                                  50435$:
 4749
 4750 023446                                   ENDAUTO
      023446                           L10013:
      023446  104461                           TRAP    C$AUTO
 4751
 4752 023450     045     101     102  AUTODM: .ASCII  /%ABUS TRAP AT %O6%N/
      023453     125     123     040
      023456     124     122     101
      023461     120     040     101
      023464     124     040     045
      023467     117     066     045
      023472     116
 4753 023473     045     101     111          .ASCIZ  /%AINTERFACE BAD OR NOT SET TO ABOVE AD%N/
      023476     116     124     105
      023501     122     106     101
      023504     103     105     040
      023507     102     101     104
      023512     040     117     122
      023515     040     116     117
      023520     124     040     123

BEQ     50440$

BIT     #TS.OFL,@TSSR(R5)
BEQ     50441$

INC     FTLCNT(R5)

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

50441$:

BR      50442$
50440$:

INC     FTLCNT(R5)

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

50442$:

50437$:

BR      50434$
50435$:

TRAP    C$AUTO
```

```
        023523      105     124     040
        023526      124     117     040
        023531      101     102     117
        023534      126     105     040
        023537      101     104     045
        023542      116     000
  4754  023544      045     101     125   NRDYM:  .ASCIZ  /%AUNIT %D1%A NOT RDY%N/
        023547      116     111     124
        023552      040     045     104
        023555      061     045     101
        023560      040     116     117
        023563      124     040     122
        023566      104     131     045
        023571      116     000
  4755                                          .EVEN
  4756
  4757                                  ;       DEVICE BUS TRAP HANDLER
  4758                                  ;       OUTPUT: TRAPD4 BYTE  1: TRAPED AT 4
  4759                                  ;                            0: NO TRAP
  4760
  4761  023574                          TRAP4:: LET TRAPD4 :B= TRAPD4 + #1                      INCB    TRAPD4
        023574  105237  003536
  4762  023600  000002                         RTI
  4763
  4764
  4765
```

```
                                .SBTTL  CLEANUP CODING SECTION

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

023602                                  BGNCLN
023602                          L$CLEAN::

023602  004737  017300                  JSR     PC,FIRSTU               ;FIND FIRST UNIT.
                                        WHILE DEVTBL(R5) NE #END DO
023606                                                                  50443$:
023606  026527  002536  177777                                          CMP     DEVTBL(R5),#END
023614  001410                                                          BEQ     50444$
023616  004737  011672                    JSR PC,WSSR                   ;WAIT FOR UNIT READY OR TIMEOUT,
023622                                    CLRVEC        TSVCT(R5)       ;RELEASE INTERRUPT VECTORS FOR ALL DEV.
023622  016500  002476                                                  MOV     TSVCT(R5),R0
023626  104436                                                          TRAP    C$CVEC
023630  004737  017346                    JSR   PC,NEXTU                ;FIND NEXT UNIT.
023634                                  ENDDO
023634  000764                                                          BR      50443$
023636                                                                  50444$:

023636                                  EXIT    CLN
023636  104432                                                          TRAP    C$EXIT
023640  000002                                                          .WORD   L10014-.

                                        .EVEN

023642                                  ENDCLN
023642                          L10014:
023642  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
                                ;--

023644                                  BGNDU
023644                          L$DU::

                                        LET R5 ;= R0 SHIFT 1            ;R5 = LOGICAL DEVICE NUMBER X 2.
023644  010005                                                                  MOV     R0,R5
023646  006305                                                                  ASL     R5
                                        LET DEVTBL(R5) ;= #NINUSE       ;SET NOT IN USE FLAG FOR THE DEVICE.
023650  012765  177774  002536                                                  MOV     #NINUSE,DEVTBL(R5)
                                        PRINTF  #DROPDM,DROPN           ;PRINT DROP DEVICE MESSAGE
023656  013746  017622                                                          MOV     DROPN,-(SP)
023662  012746  004700                                                          MOV     #DROPDM,-(SP)
023666  012746  000002                                                          MOV     #2,-(SP)
023672  010600                                                                  MOV     SP,R0
023674  104417                                                                  TRAP    C$PNTF
023676  062706  000006                                                          ADD     #6,SP
                                        EXIT    DU
023702  000167                                                                  .WORD   J$JMP
023704  000000                                                                  .WORD   L10015-2-.

                                        .EVEN

                                        ENDDU
023706                          L10015:
023706  104453                                                                  TRAP    C$DU
```

```
                                .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.
                                ;--

023710                                  BGNAU
023710                          L$AU::

                                        LET R5 := R0 SHIFT 1            ;R5 = LOGICAL DEVICE NUMBER X 2.
023710  010005                                                                  MOV     R0,R5
023712  006305                                                                  ASL     R5
                                        LET DEVTBL(R5) := R0    ;STORE UNIT # IN DEVICE TABLE.
023714  010065  002536                                                          MOV     R0,DEVTBL(R5)
                                        GPHARD  R0,R0                   ;GET HARDWARE P TABLE FROM SUPER.
023720  104442                                                                  TRAP    C$GPHRD
                                        LET TSSR(R5) := (R0)            ;SAVE TSSR ADDRESS.
023722  011065  002466                                                          MOV     (R0),TSSR(R5)
                                        LET TSDB(R5) := (R0)+ - 02      ;SAVE TSDB ADDRESS.
023726  012065  002456                                                          MOV     (R0)+,TSDB(R5)
023732  162765  000002  002456                                                  SUB     #2,TSDB(R5)
                                        LET TSVCT(R5) := (R0)           ;SAVE INTERRUPT VECTOR ADDRESS.
023740  011065  002476                                                          MOV     (R0),TSVCT(R5)
                                        SETVEC TSVCT(R5),TS4INT(R5),#INTPRI     ;SET UP INTERUPT PROCESSING CONDITIONS.
023744  012746  000340                                                          MOV     #INTPRI,-(SP)
023750  016546  002516                                                          MOV     TS4INT(R5),-(SP)
023754  016546  002476                                                          MOV     TSVCT(R5),-(SP)
023760  012746  000003                                                          MOV     #3,-(SP)
023764  104437                                                                  TRAP    C$SVEC
023766  062706  000010                                                          ADD     #10,SP
                                        LET INTFLG(R5) := #0            ;CLEAR INTERRUPT FLAGS.
023772  005065  003476                                                          CLR     INTFLG(R5)

                                        EXIT    AU
023776  000167                                                                  .WORD   .I$JMP
024000  000000                                                                  .WORD   L10016-2-.

                                        .EVEN

                                        ENDAU
024002                          L10016:
024002  104452                                                                  TRAP    C$AU

                                        ENDMOD
```

```
4887
4898                                    .TITLE HARDWARE TESTS
4899
4900                                    .SBTTL  TEST 1:  BASIC FUNCTIONS.
4901
4902                                    ;++
4903                                    ; TEST TO EXECUTE ALL TK25 FUNCTIONS.
4904                                    ;--
4905
4906 024004                                     BGNMOD
4907
4908 024004                                     BGNTST
     024004                             T1::
4909
4910 024004                                     LET RANDOM :B= #0               ;CLR THE RANDOM OPERATIONS FLAG.
     024004  105037  003521                                                     CLRB    RANDOM
4911 024010                                     LET EXPBOT :B= #0               ;CLR EXPECT BOT FLAG.
     024010  105037  003520                                                     CLRB    EXPBOT
4912
4913 024014                                     BGNSUB                          ;SUBTEST 1 - SET CHAR, DRIVE INIT, GET STATUS.
     024014                             T1.1:
     024014  104402                                                             TRAP    C$BSUB
4914
4915 024016                                     LET R2 := #BFSEQ0               ;ADR OF CMD SEQ.
     024016  012702  024642                                                     MOV     #BFSEQ0,R2
4916 024022  004737  024616                     JSR     PC,BFSEQ                ;SET UP CMD SEQ.
4917 024026  004737  006744                     JSR     PC,EXALL                ;EXECUTE CMD SEQ ON ALL DEVICES.
4918 024032  004737  017300                     JSR     PC,FIRSTU               ;FIND THE FIRST UNIT.
4919 024036                                     WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
     024036                                                                     50445$:
     024036  026527  002536  177777                                             CMP     DEVTBL(R5),#END
     024044  001434                                                             BEQ     50446$
4920 024046                                       LET R2 := MSGPKA(R5)          ;GET MSG PACKET ADR,
     024046  016502  002506                                                     MOV     MSGPKA(R5),R2
4921 024052                                       LET R2 := R2 + #12            ;GET XSTAT2 ADR.
     024052  062702  000012                                                     ADD     #12,R2
4922 024056                                       LET TS4CL(R5) := (R2) CLR.BY #177400  ;STORE CODE LEVEL FROM DTR BYTE.
     024056  011265  002526                                                     MOV     (R2),TS4CL(R5)
     024062  042765  177400  002526                                             BIC     #177400,TS4CL(R5)
4923 024070                                       IF PASCNT(R5) EQ #1 THEN      ;IF THIS IS PASS 1 THEN:
     024070  026527  003260  000001                                             CMP     PASCNT(R5),#1
     024076  001014                                                             BNE     50447$
4924 024100                                         PRINTF #CODELM,DEVTBL(R5),TS4CL(R5) ;PRINT THE TK25 MICROCODE LEVEL.
     024100  016546  002526                                                     MOV     TS4CL(R5),-(SP)
     024104  016546  002536                                                     MOV     DEVTBL(R5),-(SP)
     024110  012746  004054                                                     MOV     #CODELM,-(SP)
     024114  012746  000003                                                     MOV     #3,(SP)
     024120  010600                                                             MOV     SP,R0
     024122  104417                                                             TRAP    C$PNTF
     024124  062706  000010                                                     ADD     #10,SP
4925 024130                                       ENDIF
     024130                                                                     50447$:
4926 024130  004737  017346                       JSR   PC,NEXTU                ;FIND NEXT UNIT.
4927 024134                                     ENDDO
     024134  000740                                                             BR      50445$
     024136                                                                     50446$:
4928 024136                                     ENDSUB
```

```
                                L10020:
                                                                        TRAP    C$ESUB
4929
4930 024140                             BGNSUB                  ;SUBTEST 2 - REWIND.
     024140                     T1.2:
     024140  104402                                                    TRAP    C$BSUB
4931
4932 024142                             LET R2 := #BFSEQ1       ;ADR OF CMD SEQ.
     024142  012702  024714                                             MOV     #BFSEQ1,R2
4933 024146  004737  024616             JSR     PC,BFSEQ        ;SET UP CMD SEQ.
4934 024152  004737  006744             JSR     PC,EXALL        ;EXECUTE CMD SEQ ON ALL DEVICES.
4935 024156                             LET STAFLG :B= #0       ;CLEAR START FLAG
     024156  105037  003534                                             CLRB    STAFLG
4936 024162                             ENDSUB
     024162                     L10021:
     024162  104403                                                    TRAP    C$ESUB
4937
4938 024164                             BGNSUB                  ;SUBTEST 3 - WRITE/VERIFY.
     024164                     T1.3:
     024164  104402                                                    TRAP    C$BSUB
4939
4940 024166                             LET R2 := #BFSEQ2       ;ADR OF CMD SEQ.
     024166  012702  024726                                             MOV     #BFSEQ2,R2
4941 024172  004737  024616             JSR     PC,BFSEQ        ;SET UP CMD SEQ.
4942 024176  004737  006744             JSR     PC,EXALL        ;EXECUTE CMD SEQ ON ALL DEVICES.
4943 024202                             ENDSUB
     024202                     L10022:
     024202  104403                                                    TRAP    C$ESUB
4944
4945 024204                             BGNSUB                  ;SUBTEST 4 - WRITE TAPE MARK, ERASE.
     024204                     T1.4:
     024204  104402                                                    TRAP    C$BSUB
4946
4947 024206                             LET R2 := #BFSEQ3       ;ADR OF CMD SEQ.
     024210  012702  025020                                             MOV     #BFSEQ3,R2
4948 024212  004737  024616             JSR     PC,BFSEQ        ;SET UP CMD SEQ.
4949 024216  004737  006744             JSR     PC,EXALL        ;EXECUTE CMD SEQ ON ALL DEVICES.
4950 024222                             ENDSUB
     024222                     L10023:
     024222  104403                                                    TRAP    C$ESUB
4951
4952 024224                             BGNSUB                  ;SUBTEST 5 - SPACE FILES.
     024224                     T1.5:
     024224  104402                                                    TRAP    C$BSUB
4953
4954 024226                             LET R2 := #BFSEQ4       ;ADR OF CMD SEQ.
     024226  012702  025072                                             MOV     #BFSEQ4,R2
4955 024232  004737  024616             JSR     PC,BFSEQ        ;SET UP CMD SEQ.
4956 024236  004737  006744             JSR     PC,EXALL        ;EXECUTE CMD SEQ ON ALL DEVICES.
4957 024242                             ENDSUB
     024242                     L10024:
     024242  104403                                                    TRAP    C$ESUB
4958
4959 024244                             BGNSUB                  ;SUBTEST 6 - SPACE RECORDS.
     024244                     T1.6:
     024244  104402                                                    TRAP    C$BSUB
4960
```

```
4961 024246                                    LET R2 := #BFSEQ5              ;ADR OF CMD SEQ.
     024246  012702  025134                                                        MOV     #BFSEQ5,R2
4962 024252  004737  024616                    JSR     PC,BFSEQ               ;SET UP CMD SEQ.
4963 024256  004737  006744                    JSR     PC,EXALL               ;EXECUTE CMD SEQ ON ALL DEVICES.
4964 024262                                    ENDSUB
     024262                            L10025:
     024262  104403                                                            TRAP    C$ESUB
4965
4966 024264                                    BGNSUB                         ;SUBTEST 7 - WRITE RETRY.
     024264                            T1.7:
     024264  104402                                                            TRAP    C$BSUB
4967
4968 024266                                    LET R2 := #BFSEQ6              ;ADR OF CMD SEQ.
     024266  012702  025206                                                        MOV     #BFSEQ6,R2
4969 024272  004737  024616                    JSR     PC,BFSEQ               ;SET UP CMD SEQ.
4970 024276  004737  006744                    JSR     PC,EXALL               ;EXECUTE CMD SEQ ON ALL DEVICES.
4971 024302                                    ENDSUB
     024302                            L10026:
     024302  104403                                                            TRAP    C$ESUB
4972
4973 024304                                    BGNSUB                         ;SUBTEST 8 - READ REV RETRY.
     024304                            T1.8:
     024304  104402                                                            TRAP    C$BSUB
4974
4975 024306                                    LET R2 := #BFSEQ7              ;ADR OF CMD SEQ.
     024306  012702  025240                                                        MOV     #BFSEQ7,R2
4976 024312  004737  024616                    JSR     PC,BFSEQ               ;SET UP CMD SEQ.
4977 024316  004737  006744                    JSR     PC,EXALL               ;EXECUTE CMD SEQ ON ALL DEVICES.
4978 024322                                    ENDSUB
     024322                            L10027:
     024322  104403                                                            TRAP    C$ESUB
4979
4980 024324                                    BGNSUB                         ;SUBTEST 9 - READ FWD RETRY.
     024324                            T1.9:
     024324  104402                                                            TRAP    C$BSUB
4981
4982 024326                                    LET R2 := #BFSEQ8              ;ADR OF CMD SEQ.
     024326  012702  025272                                                        MOV     #BFSEQ8,R2
4983 024332  004737  024616                    JSR     PC,BFSEQ               ;SET UP CMD SEQ.
4984 024336  004737  006744                    JSR     PC,EXALL               ;EXECUTE CMD SEQ ON ALL DEVICES.
4985 024342                                    ENDSUB
     024342                            L10030:
     024342  104403                                                            TRAP    C$ESUB
4986
4987 024344                                    BGNSUB                         ;SUBTEST 10- CLEAN.
     024344                            T1.10:
     024344  104402                                                            TRAP    C$BSUB
4988
4989 024346                                    LET R2 := #BFSEQ9              ;ADR OF CMD SEQ.
     024346  012702  025324                                                        MOV     #BFSEQ9,R2
4990 024352  004737  024616                    JSR     PC,BFSEQ               ;SET UP CMD SEQ.
4991 024356  004737  006744                    JSR     PC,EXALL               ;EXECUTE CMD SEQ ON ALL DEVICES.
4992 024362                                    ENDSUB
     024362                            L10031:
     024362  104403                                                            TRAP    C$ESUB
4993
4994 024364                                    BGNSUB                         ;SUBTEST 11 - WTV SWAPPED DATA BYTES.
```

```
024364                          T1.11:
024364  104402                                                                        TRAP    C$BSUB

024366                                  LET R2 := #BFSE10             ;ADR OF CMD SEQ.
024366  012702  025346                                                        MOV     #BFSE10,R2
024372  004737  024616                  JSR     PC,BFSEQ              ;SET UP CMD SEQ.
024376  004737  006744                  JSR     PC,EXALL              ;WRITE/VERIFY RECORDS 1 AND 2.
024402                                  LET SWBFLG :B= #1             ;ENABLE BYTE SWAPPING.
024402  112737  000001  003524                                                MOVB    #1,SWBFLG
024410  004737  006744                  JSR     PC,EXALL              ;WRITE/VERIFY RECORDS 3 AND 4.
024414                                  LET SWBFLG :B= #0             ;DISABLE BYTE SWAPPING.
024414  105037  003524                                                        CLRB    SWBFLG
024420                                  ENDSUB
024420                          L10032:
024420  104403                                                                        TRAP    C$ESUB

024422                                  LET R2 := DATAWT + #10.       ;INIT WRITE BUFFER POINTER.
024422  013702  003414                                                        MOV     DATAWT,R2
024426  062702  000012                                                        ADD     #10.,R2
024432                                  WHILE R2 NE DATAWT DO   ;UNTIL 10 BYTES HAVE BEEN SWAPPED.
024432                                                                        50450$:
024432  020237  003414                                                        CMP     R2,DATAWT
024436  001402                                                                BEQ     50451$
024440  000342                            SWAB -(R2)                  ;SWAP DATA BYTES IN WRITE BUFFER.
024442                                  ENDDO
024442  000773                                                                BR      50450$
024444                                                                        50451$:
024444                                  LET T1SWB :B= T1SWB + #1      ;SET T1 SWAP BYTES FLAG FOR "CKDATA" SUBR
024444  105237  003530                                                        INCB    T1SWB

024450                                  BGNSUB                        ;SUBTEST 12 - READ SWAPPED DATA BYTES.
024450                          T1.12:
024450  104402                                                                        TRAP    C$BSUB

024452                                  LET CMDWRD := #RDR            ;CMD IS READ REV.
024452  012737  104401  003426                                                MOV     #RDR,CMDWRD
024460  004737  016254                  JSR     PC,VFEXC              ;VERIFY ODD LENGTH SWAP (RECORD 4).
024464                                  LET CMDPKT+CP.CNT := #12      ;CHANGE BYTE COUNT TO 10.
024464  012737  000012  002322                                                MOV     #12,CMDPKT+CP.CNT
024472  004737  016254                  JSR     PC,VFEXC              ;VERIFY EVEN LENGTH SWAP (RECORD 3).
024476                                  LET SWBFLG :B= #1             ;ENABLE BYTE SWAPPING.
024476  112737  000001  003524                                                MOVB    #1,SWBFLG
024504                                  LET CMDPKT+CP.CNT := #11                ;CHANGE BYTE COUNT TO 9.
024504  012737  000011  002322                                                MOV     #11,CMDPKT+CP.CNT
024512  004737  016254                  JSR     PC,VFEXC              ;VERIFY ODD LENGTH SWAP (RECORD 2).
024516                                  LET CMDPKT+CP.CNT := #12      ;CHANGE BYTE COUNT TO 10.
024516  012737  000012  002322                                                MOV     #12,CMDPKT+CP.CNT
024524  004737  016254                  JSR     PC,VFEXC              ;VERIFY EVEN LENGTH SWAP (RECORD 1).
024530                                  LET CMDWRD := #RDF            ;CMD IS READ FWD.
024530  012737  104001  003426                                                MOV     #RDF,CMDWRD
024536  004737  016254                  JSR     PC,VFEXC              ;VERIFY EVEN LENGTH SWAP (RECORD 1).
024542                                  LET CMDPKT+CP.CNT := #11                ;CHANGE BYTE COUNT TO 9.
024542  012737  000011  002322                                                MOV     #11,CMDPKT+CP.CNT
024550  004737  016254                  JSR     PC,VFEXC              ;VERIFY ODD LENGTH SWAP (RECORD 2).
024554                                  LET SWBFLG :B= #0             ;DISABLE BYTE SWAPPING.
024554  105037  003524                                                        CLRB    SWBFLG
024560                                  LET CMDPKT+CP.CNT := #12      ;CHANGE BYTE COUNT TO 10.
```

```
                                                                                MOV     #12,CMDPKT+CP.CNT
5027  024560  012737  000012  002322          JSR     PC,VFEXC                ;VERIFY EVEN LENGTH SWAP (RECORD 3).
5028  024566  004737  016254                  LET CMDPKT+CP.CNT := #11         ;CHANGE BYTE COUNT TO 9.
      024572                                                                    MOV     #11,CMDPKT+CP.CNT
5029  024572  012737  000011  002322          JSR     PC,VFEXC                ;VERIFY ODD LENGTH SWAP (RECORD 4).
5030
5031  024600  004737  016254                  ENDSUB
      024604                          L10033:
      024604  104403                                                          TRAP    C$ESUB
5032
5033  024606                                  LET T1SWB :B= #0                ;CLEAR T1 SWAP BYTES FLAG
      024606  105037  003530                                                  CLRB    T1SWB
5034
5035
5036  024612                                  EXIT    TST
      024612  104432                                                          TRAP    C$EXIT
      024614  000554                                                          .WORD   L10017-.
```

```
5038                                    ;       SUBROUTINE TO MOVE A COMMAND SEQUENCE TO THE SEQUENCE TABLE.
5039                                    ;       INPUTS:         R2 = FWA OF COMMAND SEQUENCE.
5040                                    ;       OUTPUTS:
5041                                    ;       REGISTERS:
5042                                    ;       CALLS:
5043
5044 024616                            BFSEQ:  LET R1 := #CMDSEQ               ;INIT SEQ TABLE ADDRESS.
     024616  012701  003542                                                             MOV     #CMDSEQ,R1
5045 024622                                    WHILE (R2) NE #END DO            ;WHILE THERE ARE MORE COMMANDS:
     024622                                                                             50452$:
     024622  021227  177777                                                             CMP     (R2),#END
     024626  001402                                                                     BEQ     50453$
5046 024630                                      LET (R1)+ := (R2)+             ;MOVE COMMANDS TO SEQ TABLE.
     024630  012221                                                                     MOV     (R2)+,(R1)+
5047 024632                                    ENDDO
     024632  000773                                                                     BR      50452$
     024634                                                                             50453$:
5048 024634                                    LET (R1) := #END                 ;STORE END OF SEQUENCE CODE.
     024634  012711  177777                                                             MOV     #END,(R1)
5049 024640  000207                            RTS     PC                       ;RETURN.
5050
5051
5052
5053                                    ;       BASIC FUNCTION COMMAND SEQUENCE
5054
5055 024642  140004                    BFSEQ0: .WORD   SCH                      ;SET CHAR. 200.         (1)
5056 024644  000200                                    200
5057 024646  000001                                    1
5058 024650  000000                                    0
5059 024652  100013                            DRI                      ;DRIVE INIT.            (2)
5060 024654  000001                            1
5061 024656  000001                            1
5062 024660  000000                            0
5063 024662  140004                            SCH                      ;SET CHAR. 20           (3)
5064 024664  000020                            20
5065 024666  000001                            1
5066 024670  000000                            0
5067 024672  100017                            GES                      ;GET STATUS.            (4)
5068 024674  000001                            1
5069 024676  000001                            1
5070 024700  000000                            0
5071 024702  140004                            SCH                      ;SET CHAR. 40.          (5)
5072 024704  000040                            40
5073 024706  000001                            1
5074 024710  000000                            0
5075 024712  177777                    .WORD   END
5076
5077 024714  102010                    BFSEQ1:         RWD                      ;REWIND TWICE.          (6)
5078 024716  000001                                    1
5079 024720  000002                                    2
5080 024722  000000                                    0
5081 024724  177777                    .WORD   END
5082
5083 024726  104105                    BFSEQ2:         WTV                      ;WRITE/VERIFY PAT 1.    (7)
5084 024730  010000                                    DATCNT
5085 024732  000001                                    1
5086 024734  000001                                    1
```

```
5087 024736  104105                          WTV             ;WTV PAT 2.         (8)
5088 024740  010000                          DATCNT
5089 024742  000001                          1
5090 024744  000002                          2
5091 024746  104105                          WTV             ;WTV PAT 3.         (9)
5092 024750  010000                          DATCNT
5093 024752  000001                          1
5094 024754  000003                          3
5095 024756  104105                          WTV             ;WTV PAT 4.         (10)
5096 024760  010000                          DATCNT
5097 024762  000001                          1
5098 024764  000004                          4
5099 024766  104105                          WTV             ;WTV PAT 5.         (11)
5100 024770  010000                          DATCNT
5101 024772  000001                          1
5102 024774  000005                          5
5103 024776  104105                          WTV             ;WTV PAT 6.         (12)
5104 025000  010000                          DATCNT
5105 025002  000001                          1
5106 025004  000006                          6
5107 025006  104105                          WTV             ;WTV PAT 0.         (13)
5108 025010  010000                          DATCNT
5109 025012  000001                          1
5110 025014  000000                          0
5111 025016  177777                  .WORD   END
5112
5113 025020  100011              BFSEQ3:     WTM             ;WRITE TAPE MARK.   (14)
5114 025022  000001                          1
5115 025024  000001                          1
5116 025026  000000                          0
5117 025030  104005                          WRT             ;WRITE 10 RECORDS.  (15)
5118 025032  010000                          DATCNT
5119 025034  000010                          10
5120 025036  000001                          1
5121 025040  100411                          ERS             ;ERASE 10 TIMES.    (16)
5122 025042  000001                          1
5123 025044  000010                          10
5124 025046  000000                          0
5125 025050  100011                          WTM             ;WRITE TAPE MARK.   (17)
5126 025052  000001                          1
5127 025054  000001                          1
5128 025056  000000                          0
5129 025060  101011                          WTR             ;WTM RETRY          (18)
5130 025062  000001                          1
5131 025064  000001                          1
5132 025066  000000                          0
5133 025070  177777                  .WORD   END
5134
5135 025072  105410              BFSEQ4:     SFR             ;SPACE 2 FILES REV. (19)
5136 025074  000002                          2
5137 025076  000001                          1
5138 025100  000000                          0
5139 025102  105010                          SFF             ;SPACE 2 FILES FWD. (20)
5140 025104  000002                          2
5141 025106  000001                          1
5142 025110  000000                          0
5143 025112  105410                          SFR             ;SPACE 2 FILES REV. (21)
```

J12


```
5144 025114  000001                              1
5145 025116  000002                              2
5146 025120  000000                              0
5147 025122  105010                              SFF             ;SPACE 2 FILES FWD.     (22)
5148 025124  000001                              1
5149 025126  000002                              2
5150 025130  000000                              0
5151 025132  177777                      .WORD   END
5152
5153 025134  102010              BFSEQ5:         RWD             ;REWIND.                (23)
5154 025136  000001                              1
5155 025140  000001                              1
5156 025142  000000                              0
5157 025144  104010                              SRF             ;SPACE 7 RECORDS FWD.   (24)
5158 025146  000007                              7
5159 025150  000001                              1
5160 025152  000000                              0
5161 025154  104410                              SRR             ;SPACE 7 RECORDS REV.   (25)
5162 025156  000007                              7
5163 025160  000001                              1
5164 025162  000000                              0
5165 025164  104010                              SRF             ;SPACE 7 RECORDS FWD.   (26)
5166 025166  000001                              1
5167 025170  000007                              7
5168 025172  000000                              0
5169 025174  104410                              SRR             ;SPACE 7 RECORDS REV.   (27)
5170 025176  000001                              1
5171 025200  000007                              7
5172 025202  000000                              0
5173 025204  177777                      .WORD   END
5174
5175 025206  102010              BFSEQ6:         RWD             ;REWIND.                (28)
5176 025210  000001                              1
5177 025212  000001                              1
5178 025214  000000                              0
5179 025216  104005                              WRT             ;WRITE.                 (29)
5180 025220  010000                              DATCNT
5181 025222  000001                              1
5182 025224  000001                              1
5183 025226  105005                              WRR             ;WRITE RETRY.           (30)
5184 025230  010000                              DATCNT
5185 025232  000001                              1
5186 025234  000001                              1
5187 025236  177777                      .WORD   END
5188
5189 025240  104401              BFSEQ7:         RDR             ;READ REV.              (31)
5190 025242  010000                              DATCNT
5191 025244  000001                              1
5192 025246  000001                              1
5193 025250  105401                              RNR             ;READ NEXT REV.         (32)
5194 025252  010000                              DATCNT
5195 025254  000001                              1
5196 025256  000001                              1
5197 025260  125401                              RNF             ;READ NEXT FWD.         (33)
5198 025262  010000                              DATCNT
5199 025264  000001                              1
5200 025266  000001                              1
```

```
5201 025270  177777                    .WORD   END
5202
5203 025272  104001          BFSEQ8:           RDF             ;READ FWD.              (34)
5204 025274  010000                            DATCNT
5205 025276  000001                            1
5206 025300  000001                            1
5207 025302  105001                            RPF             ;READ PREVIOUS FWD.     (35)
5208 025304  010000                            DATCNT
5209 025306  000001                            1
5210 025310  000001                            1
5211 025312  125001                            RPR             ;READ PREVIOUS REV.     (36)
5212 025314  010000                            DATCNT
5213 025316  000001                            1
5214 025320  000001                            1
5215 025322  177777                    .WORD   END
5216
5217 025324  101012          BFSEQ9:   .WORD   CLN             ;CLEAN.                 (37)
5218 025326  000001                            1
5219 025330  000001                            1
5220 025332  000000                            0
5221 025334  102010                            RWD             ;REWIND                 (38)
5222 025336  000001                            1
5223 025340  000001                            1
5224 025342  000000                            0
5225 025344  177777                    .WORD   END             ;END OF SEQUENCE.
5226
5227 025346  104105          BFSE10:           WTV             ;WRITE/VERIFY EVEN LENGTH.  (39)
5228 025350  000012                            12
5229 025352  000001                            1
5230 025354  000000                            0
5231 025356  104105                            WTV             ;WRITE/VERIFY ODD LENGTH.   (40)
5232 025360  000011                            11
5233 025362  000001                            1
5234 025364  000000                            0
5235 025366  177777                    .WORD   END
5236                                   .EVEN
5237
5238 025370                            ENDTST
     025370                L10017:
     025370  104401                                    TRAP    C$ETST
```

```
5240                                    .SBTTL  TEST 2: DATA RELIABILITY.
5241
5242                                    ;++
5243                                    ; TEST TO CHECK THE DATA RELIABILITY OF THE TK25.
5244                                    ;--
5245 025372                                     BGNTST
     025372                             T2::
5246
5247 025372                                     LET RANDOM :B= #1               ;SET THE RANDOM OPERATIONS FLAG.
     025372  112737  000001  003521                                             MOVB    #1,RANDOM
5248 025400                                     LET EXPBOT :B= #0               ;CLEAR EXPECT BOT FLAG.
     025400  105037  003520                                                     CLRB    EXPBOT
5249 025404                                     LET R2 := #DATCNT - #1          ;SET UP THE RECORD LENGTH MASK.
     025404  012702  010000                                                     MOV     #DATCNT,R2
     025410  005302                                                             DEC     R2
5250 025412                                     LET LENMSK := COMP R2           ;ALLOW MAXIMUM BUFFER.
     025412  010237  003436                                                     MOV     R2,LENMSK
     025416  005137  003436                                                     COM     LENMSK
5251 025422  004737  006700                     JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
5252 025426                                     IFB STAFLG NE #0 THEN           ;IF STARTING THEN:
     025426  105737  003534                                                     TSTB    STAFLG
     025432  001404                                                             BEQ     50454$
5253 025434  004737  006724                       JSR   PC,SETRW                ;CMD2=REWIND
5254 025440                                       LET STAFLG :B= #0             ;CLR START FLAG.
     025440  105037  003534                                                     CLRB    STAFLG
5255 025444                                     ENDIF
     025444                                                                     50454$:
5256 025444                                     LET (R1)+ := #WRT               ;CMD3 = WRITE.
     025444  012721  104005                                                     MOV     #WRT,(R1)+
5257 025450                                     LET (R1)+ := #DATCNT            ;SET BRF TO MAX FOR PATTERN GENERATION.
     025450  012721  010000                                                     MOV     #DATCNT,(R1)+
5258 025454                                     LET R2 := COMP #RNOPSC
     025454  012702  177740                                                     MOV     #RNOPSC,R2
     025460  005102                                                             COM     R2
5259 025462                                   LET (R1)+ := R2                   ;31 OPERATIONS.
     025462  010221                                                             MOV     R2,(R1)+
5260 025464                                   LET (R1)+ := #RANP                ;RANDOM PATTERN.
     025464  012721  000007                                                     MOV     #RANP,(R1)+
5261 025470                                   REPEAT                            ;REPEAT TO EOT:
     025470                                                                     50455$:
5262 025470                                     WHILE R1 LT #SEQEND DO          ;FILL SEQ TBL WITH RANDOM CMDS.
     025470                                                                     50456$:
     025470  020127  003632                                                     CMP     R1,#SEQEND
     025474  002012                                                             BGE     50457$
5263 025476                                       LET RANS := RANS + RANB
     025476  063737  003440  003442                                             ADD     RANB,RANS
5264 025504                                       LET R2 := RANS CLR.BY #177741  ;R2 = RANDOM # (0 - 36).
     025504  013702  003442                                                     MOV     RANS,R2
     025510  042702  177741                                                     BIC     #177741,R2
5265 025514  004772  025652                       JSR PC,@RANCMD(R2)            ;SET UP A RANDOM CMD + BRF.
5266 025520                                     ENDD
     025520  000763                                                             BR      50456$
     025522                                                                     50457$:
5267 025522                                     LET (R1) := #END                ;STORE END OF SEQUENCE CODE IN TABLE.
     025522  012711  177777                                                     MOV     #END,(R1)
5268 025526  004737  006744                     JSR   PC,EXALL                  ;GO EXECUTE ALL CMDS IN SEQUENCE TABLE.
5269 025532                                     LET R1 := #CMDSEQ               ;INIT CMD SEQ TBL POINTER.
```

```
     025532 012701 003542                                                              MOV     #CMDSEQ,R1
5270 025536                           UNTIL R2 NE #0               ;REPEAT UNTIL EOT IS REACHED
     025536 005702                                                              TST     R2
     025540 001753                                                              BEQ     50455$
5271 025542                           LET ALLEOT :B= ALLEOT + #1   ;FLAG ALL UNITS @ EOT
     025542 105237 003531                                                       INCB    ALLEOT
5272 025546 000240                    NOP
5273 025550 000240                    NOP
5274 025552 000240                    NOP
5275 025554 004737 027640             JSR     PC,T5WEOT            ;WRITE ONE RECORD BEYOND EOT ON ALL UNITS
5276                                                               ;SO THAT SHORTER READ STOP DISTANCE
5277                                                               ;SHALL POSITION HEAD IN CLEAN IRG GAP
5278                                                               ;READ REV THAT EXTRA REC TO RE-POSITION THE TAPE
5279 025560 004737 025712             JSR     PC,RANRD             ;SET UP READ REV/FWD CMDS,
5280 025564                           LET CMDSEQ+4 := COMP #RNOPSC ;# OF RECORDS FOR READ REV.
     025564 012737 177740 003546                                                MOV     #RNOPSC,CMDSEQ+4
     025572 005137 003546                                                       COM     CMDSEQ+4
5281 025576                           LET CMDSEQ+14 := CMDSEQ+4    ;# OF RECORDS FOR READ FORWARD.
     025576 013737 003546 003556                                                MOV     CMDSEQ+4,CMDSEQ+14
5282 025604                           LET (R1) := #END             ;STORE END OF SEQUENSE CODE IN SEQ TABLE.
     025604 012711 177777                                                       MOV     #END,(R1)
5283 025610 004737 006744             JSR     PC,EXALL             ;GO EXECUTE READ REV/FWD OF LAST N RECORDS.
5284 025614                           LET ALLEOT :B= #0            ;CLEAR ALL UNITS @ EOT FLAG
     025614 105037 003531                                                       CLRB    ALLEOT
5285 025620                           LET RPTFLG :B= #1            ;REQUEST PERFORMANCE REPORT DURING REWIND.
     025620 112737 000001 003523                                                MOVB    #1,RPTFLG
5286 025626                           LET R1 := #CMDSEQ            ;INIT SEQ TBL POINTER,
     025626 012701 003542                                                       MOV     #CMDSEQ,R1
5287 025632 004737 006724             JSR PC,SETRW                 ;STORE REWIND IN SEQ TBL,
5288 025636                           LET (R1) := #END             ;STORE END IN SEQ TBL,
     025636 012711 177777                                                       MOV     #END,(R1)
5289 025642 004737 006744             JSR PC,EXALL                 ;EXECUTE REWIND CMD ON ALL UNITS
5290
5291 025646                           EXIT    TST
     025646 104432                                                                      TRAP    C$EXIT
     025650 000174                                                                      .WORD   L10034-.
5292
```

```
5294                                    ;       ADDRESSES OF SUBROUTINES USED TO SET UP RANDOM OPERATIONS IN
5295                                    ;       THE DATA RELIABILITY TEST.
5296
5297 025652  025766                    RANCMD: RANWR                   ;WRITE.
5298 025654  025766                            RANWR                   ;WRITE.
5299 025656  025766                            RANWR                   ;WRITE.
5300 025660  025766                            RANWR                   ;WRITE.
5301 025662  025766                            RANWR                   ;WRITE.
5302 025664  025766                            RANWR                   ;WRITE.
5303 025666  025766                            RANWR                   ;WRITE.
5304 025670  025766                            RANWR                   ;WRITE.
5305 025672  025712                            RANRD                   ;READ.
5306 025674  025712                            RANRD                   ;READ.
5307 025676  025712                            RANRD                   ;READ.
5308 025700  025712                            RANRD                   ;READ.
5309 025702  025712                            RANRD                   ;READ.
5310 025704  025712                            RANRD                   ;READ.
5311 025706  025712                            RANRD                   ;READ.
5312 025710  025712                            RANRD                   ;READ.
5313
5314
5315
5316
5317
5318                                    ;       SUBROUTINE TO SET UP READ COMMANDS IN SEQUENCE TABLE.
5319                                    ;       INPUTS:
5320                                    ;       OUTPUTS:
5321                                    ;       REGISTERS:      R2
5322                                    ;       CALLS:
5323
5324 025712                            RANRD:  LET (R1)+ := #SRR               ;STORE SPACE RECORD REVERSE CMD
        025712  012721  104410                                                          MOV     #SRR,(R1)+
5325 025716                                     LET RANB := RANB + RANS
        025716  063737  003442  003440                                                  ADD     RANS,RANB
5326 025724                                     LET R2 := RANB CLR.BY #RNOPSC
        025724  013702  003440                                                          MOV     RANB,R2
        025730  042702  177740                                                          BIC     #RNOPSC,R2
5327 025734                                     LET (R1)+ := R2                 ;SET REPOSITION COUNT
        025734  010221                                                                  MOV     R2,(R1)+
5328 025736                                     LET (R1)+ := #1                 ;DO ONCE
        025736  012721  000001                                                          MOV     #1,(R1)+
5329 025742                                     LET (R1)+ := #RANP              ;RANDOM PATTERN.
        025742  012721  000007                                                          MOV     #RANP,(R1)+
5330 025746                                     LET (R1)+ := #RDF               ;STORE READ FWD CMD.
        025746  012721  104001                                                          MOV     #RDF,(R1)+
5331 025752                                     LET (R1)+ := #DATCNT            ;SET BRF TO MAX TO READ RANDOM LENGTHS.
        025752  012721  010000                                                          MOV     #DATCNT,(R1)+
5332 025756                                     LET (R1)+ := R2                 ;SET RANDOM # OF OPERATIONS.
        025756  010221                                                                  MOV     R2,(R1)+
5333 025760                                     LET (R1)+ := #RANP              ;RANDOM PATTERN.
        025760  012721  000007                                                          MOV     #RANP,(R1)+
5334 025764  000207                             RTS PC
```

```
5336                                    ;       SUBROUTINE TO SET UP A WRITE COMMAND IN THE SEQUENCE TABLE.
5337                                    ;       INPUTS:
5338                                    ;       OUTPUTS:
5339                                    ;       REGISTERS:
5340                                    ;       CALLS:
5341
5342 025766                            RANWR:  LET (R1)+ := #WRT               ;STORE WRITE CMD.
     025766  012721  104005                                                     MOV     #WRT,(R1)+
5343 025772  004737  026012                    JSR PC,RANW                     ;STORE BRF, # OF OPERATIONS, PATTERN.
5344 025776  000207                            RTS PC
5345
5346
5347
5348
5349
5350                                    ;       SUBROUTINE TO SET UP A WRITE/VERIFY COMMAND IN THE SEQUENCE TABLE.
5351                                    ;       INPUTS:
5352                                    ;       OUTPUTS:
5353                                    ;       REGISTERS:
5354                                    ;       CALLS:
5355
5356 026000                            RANWV:  LET (R1)+ := #WTV               ;STORE WRITE/VERIFY CMD.
     026000  012721  104105                                                     MOV     #WTV,(R1)+
5357 026004  004737  026012                    JSR PC,RANW                     ;STORE BRF, # OF OPERATIONS, PATTERN.
5358 026010  000207                            RTS     PC
5359
5360
5361
5362
5363
5364                                    ;       SUBROUTINE TO STORE BRF, # OF OPERATIONS, PATTERN IN COMMAND
5365                                    ;       SEQUENCE TABLE FOR WRITE AND WRITE/VERIFY COMMANDS.
5366                                    ;       INPUTS:
5367                                    ;       OUTPUTS:
5368                                    ;       REGISTERS:      R2
5369                                    ;       CALLS:
5370
5371 026012                            RANW:   LET (R1)+ := #DATCNT            ;SET BRF TO MAX FOR PATTERN GENERATION.
     026012  012721  010000                                                     MOV     #DATCNT,(R1)+
5372                                                                            ;RANDOM BRF WILL BE GENERATED FOR EACH RECORD.
5373 026016                                    LET RANB := RANB + RANS
     026016  063737  003442  003440                                             ADD     RANS,RANB
5374 026024                                    LET R2 := RANB CLR.BY #RNOPSC
     026024  013702  003440                                                     MOV     RANB,R2
     026030  042702  177740                                                     BIC     #RNOPSC,R2
5375 026034                                    LET (R1)+ := R2                 ;SET RANDOM # OF OPERATIONS.
     026034  010221                                                             MOV     R2,(R1)+
5376 026036                                    LET (R1)+ := #RANP              ;RANDOM PATTERN.
     026036  012721  000007                                                     MOV     #RANP,(R1)+
5377 026042  000207                            RTS PC                          ;RETURN.
5378
5379                                           .EVEN
5380
5381 026044                                    ENDTST
     026044                            L10034:
     026044  104401                                                             TRAP    C$ETST
5382
```

```
5384                                    .SBTTL  TEST 3: WRITE AND READ STREAMING TEST.
5385
5386                                    ;++
5387                                    ;
5388                                    ;       THIS TEST STREAM WRITES 9000 RECORDS OF 4096 BYTES EACH.
5389                                    ;       DATA IS THEN VERIFIED BY PERFORMING A REWIND OPERATION, FOLLOWED
5390                                    ;       BY A READ FORWARD OPERATON FOR THE 9000 RECORDS.
5391                                    ;
5392                                    ;--
5393
5394 026046                            BGNTST
     026046                            T3::
5395
5396 026046                            IFB HERE EQ #0 THEN
     026046  105737  003340                                                  TSTB    HERE
     026052  001012                                                          BNE     50460$
5397 026054                             PRINTF #TAPCAP
     026054  012746  021571                                                  MOV     #TAPCAP,-(SP)
     026060  012746  000001                                                  MOV     #1,-(SP)
     026064  010600                                                          MOV     SP,R0
     026066  104417                                                          TRAP    C$PNTF
     026070  062706  000004                                                  ADD     #4,SP
5398 026074                             LET HERE :B= HERE + #1
     026074  105237  003340                                                  INCB    HERE
5399 026100                            ENDIF
     026100
5400 026100                            LET RANDOM :B= #0               ;CLEAR THE RANDOM OPERATIONS FLAG.
     026100  105037  003521                                                  CLRB    RANDOM
5401 026104                            LET EXPBOT :B= #0               ;CLEAR THE EXPECT BOT FLAG.
     026104  105037  003520                                                  CLRB    EXPBOT
5402 026110  004737  006700            JSR PC, SETCH                   ;SET CHARACTERISTICS.
5403 026114  004737  006724            JSR PC, SETRW                   ;SET REWIND COMMAND IN BUFFER.
5404 026120                            LET STAFLG :B= #0               ;CLEAR THE START FLAG.
     026120  105037  003534                                                  CLRB    STAFLG
5405 026124                            LET (R1) := #END                ;PLACE END FLAG IN SEQUENCE TABLE.
     026124  012711  177777                                                  MOV     #END,(R1)
5406 026130  004737  006744            JSR PC, EXALL                   ;REWIND ALL UNITS.
5407
5408 026134  004737  017300            JSR PC, FIRSTU                  ;FIND THE FIRST UNIT TO TEST (OUT)
5409                                    ; ***************************************************************
5410
5411                                    ;WRITE AND READ EACH UNIT IN TURN BEFORE GOING ON TO THE NEXT.
5412
5413                                    ; ***************************************************************
5414
5415
5416 026140                            WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
     026140                                                                  50461$:
     026140  026527  002536  177777                                          CMP     DEVTBL(R5),#END
     026146  001544                                                          BEQ     50462$
5417
5418 026150                            LET BTPT := BTADDR(R5)          ;CLEAR BAD SPOT COUNTER
     026150  016537  002550  003516                                          MOV     BTADDR(R5),BTPT
5419 026156                            LET @BTPT := #0                 ;START FROM BOT
     026156  005077  155334                                                  CLR     @BTPT
5420 026162                            LET STREAM :B= #255.            ;SET FLAG - WE'RE GOING TO STREAM
     026162  112737  000377  003532                                          MOVB    #255.,STREAM
```

```
5421 026170
     026170  012701  003542                  LET R1 := @CMDSEQ              ;SETUP SEQUENCE TABLE ADDRESS
                                                                                MOV     @CMDSEQ,R1
5422 026174
     026174  012721  104005                  LET (R1)+ := @WRT              ;WRITE COMMAND
                                                                                MOV     @WRT,(R1)+
5423 026200
     026200  012721  010000                  LET (R1)+ := @DATCNT  ;4096-BYTE RECORD LENGTH.
                                                                                MOV     @DATCNT,(R1)+
5424 026204
     026204  012721  021450                  LET (R1)+ := @9000.   ;WRITE 9000 RECORDS.
                                                                                MOV     @9000.,(R1)+
5425 026210
     026210  012721  000005                  LET (R1)+ := @5                ;GENERATE AND WRITE PATTERN 5.
                                                                                MOV     @5,(R1)+
5426 026214
     026214  012711  177777                  LET (R1)  := @END              ;SET END OF SEQUENCE TABLE.
                                                                                MOV     @END,(R1)
5427
5428 026220
     026220  012701  003542                  LET R1 := @CMDSEQ              ;SEQ. TABLE ADDRESS FOR SUBR. 'SETUP'.
                                                                                MOV     @CMDSEQ,R1
5429 026224  004737  007706                  JSR PC, SETUP          ;SETUP THE COMMAND TABLE
5430 026230                                  LET R5SAVE := R5               ;SAVE R5, EH?
     026230  010537  003460                                                     MOV     R5,R5SAVE
5431
5432 026234                                    WHILE NCNT LT NCNT1 DO       ;WHILE MORE RECORDS SHOULD BE WRITTEN:
     026234                                                                 50463$:
     026234  023737  003420  003422                                          CMP     NCNT,NCNT1
     026242  002022                                                         BGE     50464$
5433
5434 026244  004737  007600                      JSR PC, CMDAC              ;SAVE ASCII COMMAND FOR ERROR MESSAGE
5435 026250  004737  010654                      JSR PC, EXCUTE             ;ISSUE COMMAND TO UNIT.
5436 026254  004737  011274                      JSR PC, GOWAIT             ;GO WAIT FOR DONE TO SET
5437 026260                                  IF DEVTBL(R5) EQ @NINUSE THEN
     026260  026527  002536  177774                                          CMP     DEVTBL(R5),@NINUSE
     026266  001005                                                         BNE     50465$
5438 026270                                  LET NCNT := NCNT1 - @1
     026270  013737  003422  003420                                          MOV     NCNT1,NCNT
     026276  005337  003420                                                 DEC     NCNT
5439 026302                                  ENDIF
     026302                                                                 50465$:
5440 026302                                       LET NCNT := NCNT + @1     ;UPDATE THE RECORD COUNT
     026302  005237  003420                                                 INC     NCNT
5441 026306                                    ENDDO                        ;END OF RECORD 'DO' LOOP
     026306  000752                                                         BR      50463$
     026310                                                                 50464$:
5442 026310                                  LET R1 := @CMDSEQ              ;RESET R1 TO TOP OF TABLE
     026310  012701  003542                                                 MOV     @CMDSEQ,R1
5443 026314  004737  006724                  JSR     PC,SETRW               ;ISSUE REWIND
5444 026320                                  LET (R1) := @END               ;PLACE END FLAG IN SEQUENCE TABLE
     026320  012711  177777                                                 MOV     @END,(R1)
5445 026324                                  LET -(SP) := L$LUN             ;SAVE THE CURRENT LUN
     026324  013746  002074                                                 MOV     L$LUN,-(SP)
5446 026330  004737  006744                  JSR     PC,EXALL                       ;DO REWIND, NOW
5447 026334                                  LET R5 := R5SAVE               ;RESTORE R5
     026334  013705  003460                                                 MOV     R5SAVE,R5
5448 026340                                  LET L$LUN := (SP)+             ;RESTORE THE CURRENT LUN
     026340  012637  002074                                                 MOV     (SP)+,L$LUN
5449 026344                                  IF DEVTBL(R5) NE @NINUSE THEN
     026344  026527  002536  177774                                          CMP     DEVTBL(R5),@NINUSE
     026352  001431                                                         BEQ     50466$
5450 026354                                   LET R1 := @CMDSEQ             ;TOP OF COMMAND TABLE
     026354  012701  003542                                                 MOV     @CMDSEQ,R1
```

```
5451 026360                                        LET (R1)+ := @RDF              ;READ FORWARD COMMAND
     026360  012721  104001                                                               MOV     @RDF,(R1)+
5452 026364                                        LET (R1)+ := @DATCNT           ;4096 BYTE RECORDS
     026364  012721  010000                                                               MOV     @DATCNT,(R1)+
5453 026370                                        LET (R1)+ := @9000.            ;9000 ITERATIONS
     026370  012721  021450                                                               MOV     @9000.,(R1)+
5454 026374                                        LET (R1)+ := @5                        ;READ PATTERN NUMBER 5
     026374  012721  000005                                                               MOV     @5,(R1)+
5455 026400                                        LET (R1)+ := @END              ;TABLE TERMINATOR
     026400  012721  177777                                                               MOV     @END,(R1)+
5456 026404                                        LET R1 := @CMDSEQ              ;TOP OF TABLE, AGAIN!
     026404  012701  003542                                                               MOV     @CMDSEQ,R1
5457 026410  004737  007706                        JSR PC, SETUP                  ;SET UP THE COMMAND TABLE
5458 026414                                        LET VFYFLG :B= @1              ;ALLOW THE DATA VERIFY
     026414  112737  000001  003522                                                       MOVB    @1,VFYFLG
5459 026422                                        LET R5SAVE := R5               ;SAVE R5
     026422  010537  003460                                                               MOV     R5,R5SAVE
5460 026426  004737  016142                        JSR PC, VFYDAT                 ;GO OFF AND CHECK READ OPERATIONS
5461 026432                                        LET R5 := R5SAVE               ;RESTORE R5
     026432  013705  003460                                                               MOV     R5SAVE,R5
5462 026436                                        ENDIF
     026436                                                                         50466$:
5463 026436                                        LET NCNT := @0                 ;CLEAR RECORD COUNT
     026436  005037  003420                                                               CLR     NCNT
5464 026442                                        LET VFYFLG :B= @0              ;CLEAR VERIFY FLAG
     026442  105037  003522                                                               CLRB    VFYFLG
5465 026446                                        LET EXPBOT :B= @0              ;CLEAR EXPECT BOT FLAG.
     026446  105037  003520                                                               CLRB    EXPBOT
5466 026452  004737  017346                        JSR PC, NEXTU                  ;GET NEXT UNIT TO TEST (UUT).
5467
5468 026456                                        ENDDO                          ;END OF UUT LOOP
     026456  000630                                                                       BR      50461$
     026460                                                                         50462$:
5469
5470 026460                                        LET STREAM :B= @0              ;CLEAR STREAMING FLAG FOR OTHER TESTS.
     026460  105037  003532                                                               CLRB    STREAM
5471 026464                                        LET ALLEOT :B= @0              ;RESET THE UNITS @ EOT STATUS
     026464  105037  003531                                                               CLRB    ALLEOT
5472 026470                                        LET RPTFLG :B= @1              ;REQUEST A REPORT
     026470  112737  000001  003523                                                       MOVB    @1,RPTFLG
5473 026476                                        LET R1 := @CMDSEQ              ;TOP OF TABLE
     026476  012701  003542                                                               MOV     @CMDSEQ,R1
5474 026502  004737  006724                        JSR PC, SETRW                  ;STORE THE REWIND COMMAND
5475 026506                                        LET (R1) := @END               ;TERMINATOR
     026506  012711  177777                                                               MOV     @END,(R1)
5476 026512  004737  006744                        JSR PC, EXALL                  ;REWIND AND REPORT STATUS FOR ALL UNITS
5477
5478 026516                                        EXIT    TST                    ;EXIT TEST
     026516  104432                                                                       TRAP    C$EXIT
     026520  000002                                                                       .WORD   L10035-.
5479                                               .EVEN                          ;JUST IN CASE.
5480
5481 026522                                        ENDTST
     026522                               L10035:
     026522  104401                                                                       TRAP    C$ETST
```

```
5483
5484                                    .SBTTL  TEST 4: WRITE COMPATABILITY/WRITE UTILITY.
5485
5486                                    ;**
5487                                    ; TEST TO WRITE RECORDS FROM BOT TO EOT.
5488                                    ;--
5489
5490 026524                                    BGNTST
     026524                             T4::
5491
5492 026524                                    LET RANDOM :B= #1                ;SET THE RANDOM OPERATIONS FLAG.
     026524  112737  000001  003521                                                    MOVB    #1,RANDOM
5493 026532                                    LET EXPBOT :B= #0                ;CLEAR EXPECT BOT FLAG.
     026532  105037  003520                                                            CLRB    EXPBOT
5494 026536                                    LET R2 := #DATCNT - #1           ;SET UP THE RECORD LENGTH MASK.
     026536  012702  010000                                                            MOV     #DATCNT,R2
     026542  005302                                                                    DEC     R2
5495 026544                                    LET LENMSK := COMP R2            ;ALLOW MAXIMUM BUFFER.
     026544  010237  003436                                                            MOV     R2,LENMSK
     026550  005137  003436                                                            COM     LENMSK
5496 026554  004737  006700                    JSR PC,SETCH                     ;CMD 1 = SET CHARACTERISTIC.
5497 026560  004737  006724                    JSR PC,SETRW                     ;CMD2=REWIND
5498 026564                                    LET STAFLG :B= #0                ;CLEAR START FLAG
     026564  105037  003534                                                            CLRB    STAFLG
5499 026570                                    REPEAT                           ;REPEAT TO EOT.
     026570                                                                            50467$:
5500 026570                                      WHILE R1 LT #SEQEND DO         ;WHILE THERE IS MORE ROOM IN SEQ TABLE:
     026570                                                                            50470$:
     026570  020127  003632                                                            CMP     R1,#SEQEND
     026574  002003                                                                   BGE     50471$
5501 026576  004737  025766                      JSR PC,RANWR                     ;STORE A WRITE CMD IN SEQUENCE TABLE.
5502 026602                                    ENDDO
     026602  000772                                                                   BR      50470$
     026604                                                                            50471$:
5503 026604                                    LET (R1) := #END                 ;STORE END OF SEQUENCE CODE IN TABLE.
     026604  012711  177777                                                            MOV     #END,(R1)
5504 026610  004737  006744                    JSR   PC,EXALL                   ;EXECUTE ALL CMDS IN SEQ TBL ON UNITS.
5505 026614                                    LET R1 := #CMDSEQ                ;INIT SEQ TBL POINTER.
     026614  012701  003542                                                            MOV     #CMDSEQ,R1
5506 026620                                    UNTIL R2 NE #0                   ;REPEAT UNTIL EOT IS REACHED
     026620  005702                                                                    TST     R2
     026622  001762                                                                   BEQ     50467$
5507 026624                                    LET ALLEOT :B= ALLEOT + #1       ;SET ALL UNITS @ EOT FLAG
     026624  105237  003531                                                            INCB    ALLEOT
5508 026630  000240                            NOP
5509 026632  000240                            NOP
5510 026634  000240                            NOP
5511 026636  004737  027640                    JSR     PC,T5WEOT                ;WRITE ONE RECORD BEYOND EOT ON ALL UNITS
5512                                                                             ;SO THAT SHORTER READ STOP DISTANCE
5513                                                                             ;SHALL POSITION HEAD IN CLEAN IRG GAP
5514                                                                             ;READ REV THAT EXTRA REC TO RE-POSITION TAPE
5515 026642                                    LET ALLEOT :B= #0                ;CLEAR ALL UNITS @ EOT FLAG
     026642  105037  003531                                                            CLRB    ALLEOT
5516 026646  004737  006724                    JSR PC,SETRW                     ;STORE REWIND IN SEQ TBL.
5517 026652                                    LET (R1) := #END                 ;STORE END IN SEQ TBL.
     026652  012711  177777                                                            MOV     #END,(R1)
5518 026656  004737  006744                    JSR PC,EXALL                     ;EXECUTE REWIND CMD ON ALL UNITS
```

```
5519
5520 026662                                    EXIT    TST
     026662  104432                                                TRAP    C$EXIT
     026664  000002                                                .WORD   L10036-.
5521
5522                                           .EVEN
5523
5524 026666                                    ENDTST
     026666                              L10036:
     026666  104401                                                TRAP    C$ETST
5525
```

```
5527
5528                                    .SBTTL  TEST 5:  READ COMPATABILITY/READ UTILITY.
5529
5530                                    ;++
5531                                    ; TEST TO READ ENTIRE TAPE FORWARD AND REVERSE.
5532                                    ;--
5533
5534 026670                                     BGNTST
     026670                             T5::
5535
5536 026670                                     LET RANDOM :B= #1               ;SET THE RANDOM OPERATIONS FLAG.
     026670  112737  000001  003521                                                 MOVB    #1,RANDOM
5537 026676                                     LET EXPBOT :B= #1               ;SET EXPECT BOT FLAG.
     026676  112737  000001  003520                                                 MOVB    #1,EXPBOT
5538 026704  004737  006700                     JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
5539 026710  004737  006724                     JSR PC,SETRW                    ;CMD2=REWIND.
5540 026714                                     LET STAFLG :B= #0               ;CLEAR START FLAG
     026714  105037  003534                                                     CLRB    STAFLG
5541 026720                                     LET (R1)+ := #RDF               ;CMD3 = READ FORWARD.
     026720  012721  104001                                                     MOV     #RDF,(R1)+
5542 026724                                     LET (R1)+ := #DATCNT            ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
     026724  012721  010000                                                     MOV     #DATCNT,(R1)+
5543 026730                                     LET (R1)+ := #77777             ;SET RECORD COUNT TO MAX FOR WHOLE TAPE.
     026730  012721  077777                                                     MOV     #77777,(R1)+
5544 026734                                     LET (R1)+ := #RANP              ;PATTERN = RANDOM.
     026734  012721  000007                                                     MOV     #RANP,(R1)+
5545 026740                                     LET (R1) := #END                ;STORE END OF SEQUENCE CODE IN TABLE.
     026740  012711  177777                                                     MOV     #END,(R1)
5546 026744  004737  006744                     JSR     PC,EXALL                ;EXECUTE ALL CMDS IN SEQ TBL ON ALL UNITS.
5547                                    ;        LET ALLEOT :B= ALLEOT + #1      ;FLAG TO ALLOW ALL UNITS AT EOT TO READ REV
5548                                    ;        LET R1 := #CMDSEQ               ;INIT CMD SEQ TBL POINTER.
5549                                    ;        LET (R1)+ := #RDR               ;CMD1 = READ REVERSE.
5550                                    ;        LET (R1)+ := #DATCNT            ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
5551                                    ;        LET (R1)+ := #77777             ;RECORD COUNT = MAX FOR WHOLE TAPE.
5552                                    ;        LET (R1)+ := #RANP              ;PATTERN = RANDOM.
5553                                    ;        LET (R1) := #END                ;STORE END OF SEQUENCE CODE IN TABLE.
5554                                    ;        JSR     PC,EXALL                ;GO EXECUTE READ REV. OF ENTIRE TAPE.
5555 026750                                     LET ALLEOT :B= #0               ;CLEAR ALL UNITS @ EOT FLAG
     026750  105037  003531                                                     CLRB    ALLEOT
5556 026754                                     LET R1 := #CMDSEQ               ;TOP OF COMMAND TABLE
     026754  012701  003542                                                     MOV     #CMDSEQ,R1
5557 026760  004737  006724                     JSR     PC,SETRW                ;ISSUE A REWIND COMMAND (TWR)
5558 026764                                     LET (R1)+ := #END               ;TERMINATOR
     026764  012721  177777                                                     MOV     #END,(R1)+
5559 026770  004737  006744                     JSR     PC,EXALL                ;ISSUE THE REWIND
5560
5561 026774                                     EXIT    TST
     026774  104432                                                             TRAP    C$EXIT
     026776  000002                                                             .WORD   L10037-.
5562
5563                                            .EVEN
5564
5565 027000                                     ENDTST
     027000                             L10037:
     027000  104401                                                             TRAP    C$ETST
```

```
5567
5568                                    .SBTTL  TEST 6: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.
5569
5570                                    ;++
5571                                    ; TEST TO EXECUTE  OPERATOR SELECTED COMMAND SEQUENCE.
5572                                    ;--
5573
5574 027002                                     BGNTST
     027002                             T6::
5575
5576 027002                                     LET RANDOM :B= #0               ;CLEAR RAMDOM MODE FLAG.
     027002  105037  003521                                                       CLRB    RANDOM
5577 027006                                     LET EXPBOT :B= #1               ;SET EXPECT BOT FLAG.
     027006  112737  000001  003520                                             MOVB    #1,EXPBOT
5578 027014                                     LET IRE :B= PIRE                ;MOVE INHIBIT RFC ERROR REPORT FLAG.
     027014  113737  002213  003526                                             MOVB    PIRE,IRE
5579 027022  004737  006700                     JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
5580 027026                                     LET CMDSEQ+2 := CHAR            ;MOVE CHAR CODE FROM P TBL TO SEQ TBL.
     027026  013737  002220  003544                                             MOV     CHAR,CMDSEQ+2
5581 027034                                     LET R2 := #CMDD                 ;R2 POINTS TO CMD2 IN SOFT P TABLE.
     027034  012702  002222                                                     MOV     #CMDD,R2
5582 027040  004737  027616                     JSR     PC,PTCMDS               ;MOVE CMD 2 FROM P TBL TO SEQ TBL.
5583 027044  004737  027616                     JSR     PC,PTCMDS               ;MOVE CMD 3 FROM P TBL TO SEQ TBL.
5584 027050  004737  027616                     JSR     PC,PTCMDS               ;MOVE CMD 4 FROM P TBL TO SEQ TBL.
5585 027054  004737  027616                     JSR     PC,PTCMDS               ;MOVE CMD 5 FROM P TBL TO SEQ TBL.
5586 027060  004737  027616                     JSR     PC,PTCMDS               ;MOVE CMD 6 FROM P TBL TO SEQ TBL.
5587 027064  004737  027616                     JSR     PC,PTCMDS               ;MOVE CMD 7 FROM P TBL TO SEQ TBL.
5588 027070  004737  027616                     JSR     PC,PTCMDS               ;MOVE END CMD FROM P TBL TO SEQ TBL.
5589 027074                                     LET JLOOP := #0                 ;CLEAR JMP CMD LOOP COUNT.
     027074  005037  003450                                                     CLR     JLOOP
5590 027100                                     LET STAFLG :B= #0               ;CLEAR START FLAG
     027100  105037  003534                                                     CLRB    STAFLG
5591 027104                                     LET R1 := #CMDSEQ               ;INIT SEQUENCE TABLE POINTER.
     027104  012701  003542                                                     MOV     #CMDSEQ,R1
5592
5593            000000          $BRJMP=0        ;ENABLE JMP SUBSTITUTION FOR BR, IF NECESSARY.
5594
5595 027110                     3$:     WHILE (R1) NE #END DO           ;WHILE THERE ARE CMDS LEFT IN SEQUENCE TBLE:
     027110                                                                 50472$:
     027110  021127  177777                                                 CMP     (R1),#END
     027114  001002                                                         BNE     .+6
     027116  000137  027560                                                 JMP     50473$
5596 027122  022711  000040                       CMP   #JMP.C,(R1)             ;IS THIS A JUMP CMD?
5597 027126  001024                               BNE   6$                      ;BR IF NOT.
5598 027130                                       LET R1 := R1 + #2             ;POINT TO BRF.
     027130  062701  000002                                                     ADD     #2,R1
5599 027134  012137  003452                       MOV   (R1)+,JLOC              ;SAVE BRF (LOCATION).
5600 027140  022137  003450                       CMP (R1)+,JLOOP               ;HAS LOOP COUNT BE SATISFIED?
5601 027144  001003                               BNE   1$                      ;IF NOT, JMP AGAIN.
5602 027146                                       LET R1 := R1 + #2             ;IF SO, ADJUST SEQ POUNTER
     027146  062701  000002                                                     ADD     #2,R1
5603 027152  000756                               BR    3$                      ;AND GO TO NEXT COMMAND.
5604 027154                     1$:               LET JLOOP := JLOOP + #1       ;UPDATE THE LOOP COUNT.
     027154  005237  003450                                                     INC     JLOOP
5605 027160                                       LET R1 := #CMDSEQ             ;INIT CMD SEQ TABLE POINTER.
     027160  012701  003542                                                     MOV     #CMDSEQ,R1
5606 027164  005337  003452                       DEC   JLOC                    ;DECR LOCATION COUNTER.
```

```
5607 027170  001747                   BEQ   3$                    ;IF THIS IS THE RIGHT LOCATION TO JMP TO, GO SET UP.
5608 027172                           LET R1 := R1 + #10          ;IF NOT, UPDATE SEQ POINTER TO NEXT CMD.
     027172  062701  000010                                                 ADD     #10,R1
5609 027176  000772                   BR    2$                    ;DO IT AGAIN.
5610 027200  022711  000020     6$:   CMP   #DLY.C,(R1)           ;DELAY?
5611 027204  001026                   BNE   4$                    ;BR IF NOT.
5612 027206                           LET R1 := R1 + #4           ;R1 - LOCATION OF N COUNT.
     027206  062701  000004                                                 ADD     #4,R1
5613 027212                           LET TIME2 := (R1)           ;SAVE N COUNT.
     027212  011137  003446                                                 MOV     (R1),TIME2
5614 027216                     7$:   DELAY 10.                   ;GO TO SUPER-WAIT 1 MSEC.
     027216  012727  000012                                                     MOV     #10.,(PC)+
     027222  000000                                                             .WORD   0
     027224  013727  002116                                                     MOV     L$DLY,(PC)+
     027230  000000                                                             .WORD   0
     027232  005367  177772                                                     DEC     -6(PC)
     027236  001375                                                             BNE     .-4
     027240  005367  177756                                                     DEC     -22(PC)
     027244  001367                                                             BNE     .-20
5615 027246  005337  003446           DEC   TIME2
5616 027252  001361                   BNE   7$
5617 027254                           LET R1 := R1 + #4           ;POINT TO NEXT CMD.
     027254  062701  000004                                                 ADD     #4,R1
5618 027260  000713                   BR    3$                    ;GO CHECK NEXT CMD.
5619 027262  004737  007706     4$:   JSR   PC,SETUP              ;GO SETUP THE COMMAND BLOCK.
5620 027266                           WHILE NCNT LT NCNT1 DO      ;WHILE THERE ARE RECORDS REMAINING:
     027266                                                                 50474$:
     027266  023737  003420  003422                                         CMP     NCNT,NCNT1
     027274  002402                                                         BLT     .+6
     027276  000137  027550                                                 JMP     50475$
5621 027302  004737  007600             JSR PC,CMDAC                ;STORE CMD ASCII IN ERROR MSG.
5622 027306  004737  007240             JSR PC,EXSUB                ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATUS.
5623 027312                             IF CMDWRD EQ #GES THEN      ;IF CMD IS GET STATUS THEN:
     027312  023727  003426  100017                                         CMP     CMDWRD,#GES
     027320  001402                                                         BEQ     .+6
     027322  000137  027332                                                 JMP     50476$
5624 027326  004737  017624               JSR PC,PRXST              ;PRINT EXTENDED STATUS REGISTERS.
5625 027332                             ENDIF
     027332                                                                 50476$:
5626 027332  004737  017706             JSR PC,CKHAF                ;CHECK HALT AFTER EACH CMD FLAG.
5627 027336                             LET R2 := #1                ;SET ALL UNITS AT BOT/EOT.
     027336  012702  000001                                                 MOV     #1,R2
5628 027342  004737  017300             JSR PC,FIRSTU               ;FIND FIRST UNIT.
5629 027346                             WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE UNITS:
     027346                                                                 50477$:
     027346  026527  002536  177777                                         CMP     DEVTBL(R5),#END
     027354  001002                                                         BNE     .+6
     027356  000137  027454                                                 JMP     50500$
5630 027362                               IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
     027362  032737  000400  003426                                         BIT     #MOD.CO,CMDWRD
     027370  001002                                                         BNE     .+6
     027372  000137  027420                                                 JMP     50501$
5631 027376                                 IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT THEN:
     027376  032765  000002  003506                                         BIT     #XO.BOT,EOTFLG(R5)
     027404  001402                                                         BEQ     .+6
     027406  000137  027414                                                 JMP     50502$
5632 027412                                   LET R2 := #0              ;CLEAR EOT/BOT FLAG.
```

```
     027412  005002                                                                 CLR     R2
5633 027414                                 ENDIF
     027414                                                                         50502$:
5634 027414                               ELSE                      ;ELSE IF CMD IS NOT REVERSE:
     027414  000137  027446                                                         JMP     50503$
     027420                                                                         50501$:
5635 027420                                 IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
     027420  032765  000001  003506                                                 BIT     #XO.EOT,EOTFLG(R5)
     027426  001406                                                                 BEQ     50504$
     027430  032737  000001  003426                                                 BIT     #CMD.CO,CMDWRD
     027436  001402                                                                 BEQ     .+6
     027440  000137  027446                                                         JMP     50505$
     027444                                                                         50504$:
5636                                                                ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
5637 027444                                   LET R2 := #0            ;CLEAR EOT/BOT FLAG.
     027444  005002                                                                 CLR     R2
5638 027446                                 ENDIF
     027446                                                                         50505$:
5639 027446                               ENDIF
     027446                                                                         50503$:
5640 027446  004737  017346                JSR PC,NEXTU             ;FIND NEXT UNIT
5641 027452                             ENDDO                       ;
     027452  000735                                                                 BR      50477$
     027454                                                                         50500$:
5642 027454                             IF R2 EQ #1 THEN            ;IF ALL UNIT ARE AT EOT/BOT THEN:
     027454  020227  000001                                                         CMP     R2,#1
     027460  001402                                                                 BEQ     .+6
     027462  000137  027530                                                         JMP     50506$
5643 027466                               LET NCNT1 := NCNT + #1    ;FORCE TERMINATION OF COMMAND.
     027466  013737  003420  003422                                                 MOV     NCNT,NCNT1
     027474  005237  003422                                                         INC     NCNT1
5644 027500                               LET ALLEOT :B= ALLEOT + #1 ;FLAG ALL UNITS AT EOT/BOT TO ALLOW VERIFY OF DATA
     027500  105237  003531                                                         INCB    ALLEOT
5645 027504                               IF CMDLG EQ #2 THEN       ;WHEN WRITING IS CURRENT COMMAND
     027504  023727  003434  000002                                                 CMP     CMDLG,#2
     027512  001402                                                                 BEQ     .+6
     027514  000137  027524                                                         JMP     50507$
5646 027520  004737  027640                  JSR PC,T5WEOT          ;GO WRITE/READ REV ONE RECORD BEYOND EOT
5647 027524                               ENDIF
     027524                                                                         50507$:
5648 027524                             ELSE
     027524  000137  027534                                                         JMP     50510$
     027530                                                                         50506$:
5649 027530                               LET ALLEOT :B= #0 ;WHEN NOT ALL @EOT, CLEAR FLAG
     027530  105037  003531                                                         CLRB    ALLEOT
5650 027534                             ENDIF
     027534                                                                         50510$:
5651 027534                             LET NCNT := NCNT + #1       ;UPDATE RECORD COUNT.
     027534  005237  003420                                                         INC     NCNT
5652 027540                             LET PCMDWD := CMDWRD        ;SAVE PREVIOUS COMMAND WORD.
     027540  013737  003426  003432                                                 MOV     CMDWRD,PCMDWD
5653 027546                           ENDDO
     027546  000647                                                                 BR      50474$
     027550                                                                         50475$:
5654 027550  004737  016142          JSR    PC,VFYDAT               ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
5655                                                                ;VERIFY THE LAST N RECORDS OF DATA.
5656
```

```
5657 027554                                ENDDO
     027554  000137  027110                                                  JMP     50472$
     027560                                                                  50473$:
5658
5659         177777                $BRJMP=-1       ;TURN OFF JMP SUBSTITUTION (SPMACJ CONTROL, ONLY).
5660 027560                                LET ALLEOT :B= #0                ;NOT AT EOT
     027560  105037  003531                                                  CLRB    ALLEOT
5661 027564                                LET RPTFLG :B= #1                ;REQUEST A REPORT
     027564  112737  000001  003523                                         MOVB    #1,RPTFLG
5662 027572                                LET R1 := #CMDSEQ                ;SET UP THE COMMAND TABLE ADDRESS
     027572  012701  003542                                                  MOV     #CMDSEQ,R1
5663 027576  004737  006724                JSR PC,SETRW                     ;STORE REWIND COMMAND
5664 027602                                LET (R1) := #END                 ;TERMINATE THE TABLE
     027602  012711  177777                                                  MOV     #END,(R1)
5665 027606  004737  006744                JSR PC,EXALL                     ;ISSUE THE REWINDS
5666
5667
5668 027612                                EXIT    TST
     027612  104432                                                          TRAP    C$EXIT
     027614  000130                                                          .WORD   L10040-.
5669
```

```
5671
5672                                    ;       SUBROUTINE TO MOVE A COMMAND FROM THE SOFTWARE P TABLE TO
5673                                    ;       THE COMMAND SEQUENCE TABLE.
5674                                    ;       INPUTS:         R2 = POINTER TO SOFT "P" TABLE
5675                                    ;       OUTPUTS:
5676                                    ;       REGISTERS:      R3.
5677                                    ;       CALLS:
5678
5679 027616                            PTCMDS: LET R3 := (R2)+ - #1 SHIFT +1   ;R3 = COMMAND TABLE INDEX.
     027616  012203                                                                 MOV     (R2)+,R3
     027620  005303                                                                 DEC     R3
     027622  006303                                                                 ASL     R3
5680 027624                                     LET (R1)+ := CMDTBL(R3)         ;MOVE COMMAND WORD.
     027624  016321  003644                                                         MOV     CMDTBL(R3),(R1)+
5681 027630                                     LET (R1)+ := (R2)+              ;MOVE # OF BYTES.
     027630  012221                                                                 MOV     (R2)+,(R1)+
5682 027632                                     LET (R1)+ := (R2)+              ;MOVE # OF OPERATIONS.
     027632  012221                                                                 MOV     (R2)+,(R1)+
5683 027634                                     LET (R1)+ := (R2)+              ;MOVE PATTERN CODE.
     027634  012221                                                                 MOV     (R2)+,(R1)+
5684 027636  000207                           RTS PC
5685
5686                                    ;       SUBROUTINE TO WRITE THEN READ REVERSE ONE RECORD BEYOND EOT
5687                                    ;       INPUTS:
5688                                    ;       OUTPUTS:
5689                                    ;       REGISTERS:
5690                                    ;       CALLS:          CMDAC,EXSUB,CKHAE
5691
5692 027640  000240                    T5WEOT: NOP
5693 027642  000240                            NOP
5694 027644  004737  007240                    JSR PC,EXSUB            ;WRITE ONE RECORD BEYOND EOT
5695 027650  004737  017706                    JSR PC,CKHAE            ;SO THAT READ SHORTER STOP DISTANCE
5696                                                                    ;SHALL POSITION HEAD IN CLEAN IRG GAP
5697 027654                                     LET PCMDWD := CMDWRD    ;REPOSITION TAPE
     027654  013737  003426  003432                                                 MOV     CMDWRD,PCMDWD
5698 027662                                     LET CMDWRD := #RDR      ;BEFORE EXTRA RECORD
     027662  012737  104401  003426                                                 MOV     #RDR,CMDWRD
5699 027670                                     LET CMDLG := #4         ;BY READING REVERSE
     027670  012737  000004  003434                                                 MOV     #4,CMDLG
5700 027676                                     LET CMDPKT := CMDWRD CLR,BY #BRF,C
     027676  013737  003426  002314                                                 MOV     CMDWRD,CMDPKT
     027704  042737  004000  002314                                                 BIC     #BRF,C,CMDPKT
5701 027712                                     LET CMDSAV := CMDPKT    ;THAT RECORD TO ALLOW
     027712  013737  002314  003430                                                 MOV     CMDPKT,CMDSAV
5702 027720                                     LET CMDPKT+CP.ADL := DATARD     ;NEXT COMMAND IN THE
     027720  013737  003416  002316                                                 MOV     DATARD,CMDPKT+CP.ADL
5703 027726  004737  007600                    JSR PC,CMDAC            ;TABLE TO BE EXECUTED
5704 027732  004737  007240                    JSR PC,EXSUB
5705 027736  004737  017706                    JSR PC,CKHAE
5706 027742  000207                           RTS PC
5707                                           .EVEN
5708 027744                                    ENDTST
     027744                             L10040:
     027744  104401                                                                 TRAP    C$ETST
5709 027746                                    ENDMOD
```

```
                                .TITLE PARAMETER CODING

                                .SBTTL  HARDWARE PARAMETER CODING SECTION

027746                                  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.
                                ;-

027746                                  BGNHRD
027746  000024                                                          .WORD L10041-L$HARD/2
027750                          L$HARD::

027750                                  GPRMA   TS4ADR,0,0,160002,177564,YES
027750  000031                                                          .WORD   T$CODE
027752  027774                                                          .WORD   TS4ADR
027754  160002                                                          .WORD   T$LOLIM
027756  177564                                                          .WORD   T$HILIM
027760                                  GPRMD   TS4VCT,2,0,777,60,776,YES
027760  001032                                                          .WORD   T$CODE
027762  030011                                                          .WORD   TS4VCT
027764  000777                                                          .WORD   777
027766  000060                                                          .WORD   T$LOLIM
027770  000776                                                          .WORD   T$HILIM

027772                                  EXIT HRD
027772  013004                                                          .WORD   T$CODE

                                        .NLIST  BEX
027774      124     123     123 TS4ADR: .ASCIZ  /TSSR ADDRESS/
030011      126     105     103 TS4VCT: .ASCIZ  /VECTOR/
                                        .LIST   BEX
                                        .EVEN

030020                                  ENDHRD
                                                                        .EVEN
030020                          L10041:
```

```
                        .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.
                        ;--

                                BGNSFT

030020  000531                                                 .WORD L10042-L$SOFT/2
030022                  L$SOFT::

                                GPRML   CLRM,0,1,YES
030022  000130                                                 .WORD   T$CODE
030024  030600                                                 .WORD   CLRM
030026  000001                                                 .WORD   1
                                GPRML   RRVM,0,400,YES
030030  000130                                                 .WORD   T$CODE
030032  030617                                                 .WORD   RRVM
030034  000400                                                 .WORD   400
                                GPRML   HAEM,2,1,YES
030036  001130                                                 .WORD   T$CODE
030040  030646                                                 .WORD   HAEM
030042  000001                                                 .WORD   1
                                GPRML   RCVERM,2,400,YES
030044  001130                                                 .WORD   T$CODE
030046  030672                                                 .WORD   RCVERM
030050  000400                                                 .WORD   400
                                GPRML   IRECM,4,1,YES
030052  002130                                                 .WORD   T$CODE
030054  030714                                                 .WORD   IRECM
030056  000001                                                 .WORD   1
                                XFERT   NEXTSP
030060  004024                                                 .WORD   T$CODE
                                GPRML   BADTM,4,400,YES
030062  002130                                                 .WORD   T$CODE
030064  030735                                                 .WORD   BADTM
030066  000400                                                 .WORD   400
                        NEXTSP: GPRML   DINTM,6,1,YES
030070  003130                                                 .WORD   T$CODE
030072  030762                                                 .WORD   DINTM
030074  000001                                                 .WORD   1
                                GPRML   IREM,6,400,YES
030076  003130                                                 .WORD   T$CODE
030100  031005                                                 .WORD   IREM
030102  000400                                                 .WORD   400
                                GPRML   RAMM,10,1,YES
030104  004130                                                 .WORD   T$CODE
030106  031040                                                 .WORD   RAMM
030110  000001                                                 .WORD   1

                                GPRML   EWPT,12,1,YES
030112  005130                                                 .WORD   T$CODE
```

```
        030114  031064                                                  .WORD   EWPT
        030116  000001                                                  .WORD   1
5804    030120                  GPRML   CHGM,12,400,YES
        030120  005130                                                  .WORD   T$CODE
        030122  031122                                                  .WORD   CHGM
        030124  000400                                                  .WORD   400
5805    030126                  XFERF   ENDSP1
        030126  127044                                                  .WORD   T$CODE
5806    030130                  GPRMO   CHARM,14,0,377,0,777,YES
        030130  006032                                                  .WORD   T$CODE
        030132  031141                                                  .WORD   CHARM
        030134  000377                                                  .WORD   377
        030136  000000                                                  .WORD   T$LOLIM
        030140  000777                                                  .WORD   T$HILIM
5807    030142                  GPRMO   CMD2M,16,D,37,1,33,YES
        030142  007052                                                  .WORD   T$CODE
        030144  031166                                                  .WORD   CMD2M
        030146  000037                                                  .WORD   37
        030150  000001                                                  .WORD   T$LOLIM
        030152  000033                                                  .WORD   T$HILIM
5808    030154                  GPRMO   BPCRM,20,D,-1,1,DATCNT,YES
        030154  010052                                                  .WORD   T$CODE
        030156  031174                                                  .WORD   BPCRM
        030160  177777                                                  .WORD   -1
        030162  000001                                                  .WORD   T$LOLIM
        030164  010000                                                  .WORD   T$HILIM
5809    030166                  GPRMO   NUMBM,22,D,-1,1,77777,YES
        030166  011052                                                  .WORD   T$CODE
        030170  031206                                                  .WORD   NUMBM
        030172  177777                                                  .WORD   -1
        030174  000001                                                  .WORD   T$LOLIM
        030176  077777                                                  .WORD   T$HILIM
5810    030200                  GPRMO   PATTM,24,D,17,0,10,YES
        030200  012052                                                  .WORD   T$CODE
        030202  031226                                                  .WORD   PATTM
        030204  000017                                                  .WORD   17
        030206  000000                                                  .WORD   T$LOLIM
        030210  000010                                                  .WORD   T$HILIM
5811    030212                  GPRMO   CMD3M,26,D,37,1,33,YES
        030212  013052                                                  .WORD   T$CODE
        030214  031240                                                  .WORD   CMD3M
        030216  000037                                                  .WORD   37
        030220  000001                                                  .WORD   T$LOLIM
        030222  000033                                                  .WORD   T$HILIM
5812    030224                  GPRMO   BPCRM,30,D,-1,1,DATCNT,YES
        030224  014052                                                  .WORD   T$CODE
        030226  031174                                                  .WORD   BPCRM
        030230  177777                                                  .WORD   -1
        030232  000001                                                  .WORD   T$LOLIM
        030234  010000                                                  .WORD   T$HILIM
5813    030236                  GPRMO   NUMBM,32,D,-1,1,77777,YES
        030236  015052                                                  .WORD   T$CODE
        030240  031206                                                  .WORD   NUMBM
        030242  177777                                                  .WORD   -1
        030244  000001                                                  .WORD   T$LOLIM
        030246  077777                                                  .WORD   T$HILIM
5814    030250                  GPRMO   PATTM,34,D,17,0,10,YES
```

```
                030250  016052                                                  .WORD   T$CODE
                030252  031226                                                  .WORD   PATTM
                030254  000017                                                  .WORD   17
                030256  000000                                                  .WORD   T$LOLIM
                030260  000010                                                  .WORD   T$HILIM
     5815       030262                  GPRMD   CMD4M,36,D,37,1,33,YES
                030262  017052                                                  .WORD   T$CODE
                030264  031246                                                  .WORD   CMD4M
                030266  000037                                                  .WORD   37
                030270  000001                                                  .WORD   T$LOLIM
                030272  000033                                                  .WORD   T$HILIM
     5816       030274                  GPRMD   BPCRM,40,D,-1,1,DATCNT,YES
                030274  020052                                                  .WORD   T$CODE
                030276  031174                                                  .WORD   BPCRM
                030300  177777                                                  .WORD   -1
                030302  000001                                                  .WORD   T$LOLIM
                030304  010000                                                  .WORD   T$HILIM
     5817       030306                  GPRMD   NUMBM,42,D,-1,1,77777,YES
                030306  021052                                                  .WORD   T$CODE
                030310  031206                                                  .WORD   NUMBM
                030312  177777                                                  .WORD   -1
                030314  000001                                                  .WORD   T$LOLIM
                030316  077777                                                  .WORD   T$HILIM
     5818       030320                  GPRMD   PATTM,44,D,17,0,10,YES
                030320  022052                                                  .WORD   T$CODE
                030322  031226                                                  .WORD   PATTM
                030324  000017                                                  .WORD   17
                030326  000000                                                  .WORD   T$LOLIM
                030330  000010                                                  .WORD   T$HILIM
     5819       030332                  GPRMD   CMD5M,46,D,37,1,33,YES
                030332  023052                                                  .WORD   T$CODE
                030334  031254                                                  .WORD   CMD5M
                030336  000037                                                  .WORD   37
                030340  000001                                                  .WORD   T$LOLIM
                030342  000033                                                  .WORD   T$HILIM
     5820       030344                  GPRMD   BPCRM,50,D,-1,1,DATCNT,YES
                030344  024052                                                  .WORD   T$CODE
                030346  031174                                                  .WORD   BPCRM
                030350  177777                                                  .WORD   -1
                030352  000001                                                  .WORD   T$LOLIM
                030354  010000                                                  .WORD   T$HILIM
     5821       030356                  GPRMD   NUMBM,52,D,-1,1,77777,YES
                030356  025052                                                  .WORD   T$CODE
                030360  031206                                                  .WORD   NUMBM
                030362  177777                                                  .WORD   -1
                030364  000001                                                  .WORD   T$LOLIM
                030366  077777                                                  .WORD   T$HILIM
     5822       030370                  GPRMD   PATTM,54,D,17,0,10,YES
                030370  026052                                                  .WORD   T$CODE
                030372  031226                                                  .WORD   PATTM
                030374  000017                                                  .WORD   17
                030376  000000                                                  .WORD   T$LOLIM
                030400  000010                                                  .WORD   T$HILIM
     5823       030402                  XFER    ENDSP2
                030402  002004                                                  .WORD   T$CODE
     5824       030404  ENDSP1: XFER    ENDSP
                030404  075004                                                  .WORD   T$CODE
```

```
5825 030406                    ENDSP2: GPRMD   CMD6M,56,D,37,1,33,YES
     030406  027052                                                    .WORD   T$CODE
     030410  031262                                                    .WORD   CMD6M
     030412  000037                                                    .WORD   37
     030414  000001                                                    .WORD   T$LOLIM
     030416  000033                                                    .WORD   T$HILIM
5826 030420                            GPRMD   BPCRM,60,D,-1,1,DATCNT,YES
     030420  030052                                                    .WORD   T$CODE
     030422  031174                                                    .WORD   BPCRM
     030424  177777                                                    .WORD   -1
     030426  000001                                                    .WORD   T$LOLIM
     030430  010000                                                    .WORD   T$HILIM
5827 030432                            GPRMD   NUMBM,62,D,-1,1,77777,YES
     030432  031052                                                    .WORD   T$CODE
     030434  031206                                                    .WORD   NUMBM
     030436  177777                                                    .WORD   -1
     030440  000001                                                    .WORD   T$LOLIM
     030442  077777                                                    .WORD   T$HILIM
5828 030444                            GPRMD   PATTM,64,D,17,0,10,YES
     030444  032052                                                    .WORD   T$CODE
     030446  031226                                                    .WORD   PATTM
     030450  000017                                                    .WORD   17
     030452  000000                                                    .WORD   T$LOLIM
     030454  000010                                                    .WORD   T$HILIM
5829 030456                            GPRMD   CMD7M,66,D,37,1,33,YES
     030456  033052                                                    .WORD   T$CODE
     030460  031270                                                    .WORD   CMD7M
     030462  000037                                                    .WORD   37
     030464  000001                                                    .WORD   T$LOLIM
     030466  000033                                                    .WORD   T$HILIM
5830 030470                            GPRMD   BPCRM,70,D,-1,1,DATCNT,YES
     030470  034052                                                    .WORD   T$CODE
     030472  031174                                                    .WORD   BPCRM
     030474  177777                                                    .WORD   -1
     030476  000001                                                    .WORD   T$LOLIM
     030500  010000                                                    .WORD   T$HILIM
5831 030502                            GPRMD   NUMBM,72,D,-1,1,77777,YES
     030502  035052                                                    .WORD   T$CODE
     030504  031206                                                    .WORD   NUMBM
     030506  177777                                                    .WORD   -1
     030510  000001                                                    .WORD   T$LOLIM
     030512  077777                                                    .WORD   T$HILIM
5832 030514                            GPRMD   PATTM,74,D,17,0,10,YES
     030514  036052                                                    .WORD   T$CODE
     030516  031226                                                    .WORD   PATTM
     030520  000017                                                    .WORD   17
     030522  000000                                                    .WORD   T$LOLIM
     030524  000010                                                    .WORD   T$HILIM
5833 030526                            GPRMD   CMD8M,76,D,37,1,33,YES
     030526  037052                                                    .WORD   T$CODE
     030530  031276                                                    .WORD   CMD8M
     030532  000037                                                    .WORD   37
     030534  000001                                                    .WORD   T$LOLIM
     030536  000033                                                    .WORD   T$HILIM
5834 030540                            GPRMD   BPCRM,100,D,-1,1,DATCNT,YES
     030540  040052                                                    .WORD   T$CODE
     030542  031174                                                    .WORD   BPCRM
```

```
        030544  177777                                                  .WORD   -1
        030546  000001                                                  .WORD   T$LOLIM
        030550  010000                                                  .WORD   T$HILIM
5835    030552                          GPRMD   NUMBM,102,D,-1,1,77777,YES
        030552  041052                                                  .WORD   T$CODE
        030554  031206                                                  .WORD   NUMBM
        030556  177777                                                  .WORD   -1
        030560  000001                                                  .WORD   T$LOLIM
        030562  077777                                                  .WORD   T$HILIM
5836    030564                          GPRMD   PATTM,104,D,17,0,10,YES
        030564  042052                                                  .WORD   T$CODE
        030566  031226                                                  .WORD   PATTM
        030570  000017                                                  .WORD   17
        030572  000000                                                  .WORD   T$LOLIM
        030574  000010                                                  .WORD   T$HILIM
5837    030576                  ENDSP:
5838    030576                          XFER    JMPMSG
        030576  120004                                                  .WORD   T$CODE
```

```
5840
5847                                            .NLIST  BEX
5848 030600     103     114     105  CLRM:     .ASCIZ  /CLEAR COUNTERS/
5849 030617     122     105     123  RRVM:     .ASCIZ  /RESET RANDOM VARIABLES/
5850 030646     110     101     114  HAEM:     .ASCIZ  /HALT AFTER EACH CMD/
5851 030672     120     122     111  RCVERM:   .ASCIZ  /PRINT SOFT ERRORS/
5852 030714     111     116     110  IRECM:    .ASCIZ  /INHIBIT RECOVERY/
5853 030735     102     101     104  BADTM:    .ASCIZ  /BAD TAPE SPOT DETECT/
5854 030762     104     111     123  DINTM:    .ASCIZ  /DISABLE INTERRUPTS/
5855 031005     111     116     110  IREM:     .ASCIZ  /INHIBIT RFC ERROR REPORT/
5856                                            .LIST   BEX
5857                                            .EVEN
5858 031036                             JMPMSG:
5859 031036                                     XFER    JMPMS2
     031036  100004                                                    .WORD   T$CODE
5860 031040     103     122     116  RAMM:     .ASCIZ  /CONTROLLER RAM DUMP/
     031043     124     122     117
     031046     114     114     105
     031051     122     040     122
     031054     101     115     040
     031057     104     122     115
     031062     120     000
5861 031064     105     116     101  EWPT:     .ASCIZ  /ENABLE EARLY WARNING MESSAGES/
     031067     102     114     105
     031072     040     105     101
     031075     122     114     123
     031100     040     127     101
     031103     122     116     111
     031106     116     107     040
     031111     115     105     123
     031114     123     101     107
     031117     105     123     000
5862 031122     103     110     101  CHGM:     .ASCIZ  /CHANGE CMD SEQ/
     031125     116     107     105
     031130     040     103     115
     031133     104     040     123
     031136     105     121     000
5863 031141     103     110     101  CHARM:    .ASCIZ  /CHARACTERISTICS CODE/
     031144     122     101     103
     031147     124     105     122
     031152     111     123     124
     031155     111     103     123
     031160     040     103     117
     031163     104     105     000
5864 031166     103     115     104  CMD2M:    .ASCIZ  "CMD/2"
     031171     057     062     000
5865 031174     102     122     106  BPCRM:    .ASCIZ  /BRF COUNT/
     031177     040     103     117
     031202     125     116     124
     031205     000
5866 031206     043     040     117  NUMBM:    .ASCIZ  /# OF OPERATIONS/
     031211     106     040     117
     031214     120     105     122
     031217     101     124     111
     031222     117     116     123
     031225     000
5867 031226     120     101     124  PATTM:    .ASCIZ  /PATTERN/
```

```
      031231     124     105     122
      031234     116     000
 5868                                 .LIST   BEX
 5869                                 .EVEN
 5870 031236                     JMPMS2:
 5871 031236                             EXIT SFT
      031236  023004                                                    .WORD   T$CODE
 5872
 5873                                 .NLIST  BEX
 5874 031240     103     115     104 CMD3M:   .ASCIZ  "CMD/3"
 5875 031246     103     115     104 CMD4M:   .ASCIZ  "CMD/4"
 5876 031254     103     115     104 CMD5M:   .ASCIZ  "CMD/5"
 5877 031262     103     115     104 CMD6M:   .ASCIZ  "CMD/6"
 5878 031270     103     115     104 CMD7M:   .ASCIZ  "CMD/7"
 5879 031276     103     115     104 CMD8M:   .ASCIZ  "CMD/8"
 5880                                 .LIST   BEX
 5881 031304                             ENDSFT
      031304                     L10042:                                 .EVEN
 5882
 5883                            ;*****************************************************************************
 5884                            ;*****************************************************************************
 5885                            ;       PATCH AREA
 5886
 5887 031304                    PATCH:: .BLKW   64.
 5888                            ;*****************************************************************************
 5889                            ;*****************************************************************************
 5890
 5891 031504                             LASTAD
      031504  031520                                                    .EVEN
      031506  000004                                                    .WORD T$FREE
      031510                                                            .WORD T$SIZE
 5892 031[illegible]        L$LAST::
                                 ENDMOD
```

```
5894                                    .SBTTL  HARD CODED P-TBL
5895
5896                                    ;++
5897                                    ;DIAG IS PRE-PARAMETERIZED PER TBL
5898                                    ;--
5899
5900 031510                                     BGNSETUP 1
5901 031510                                       BGNPTAB
     031510  000000                                                    .WORD   0
     031512  000002                                                    .WORD   L10045-./2-1
     031514                             L10043:
5902 031514  172522                                 172522
5903 031516  000224                                 224
5904 031520                                       ENDPTAB
     031520                             L10045:
5905 031520                                     ENDSETUP
5906
5907         000001                     .END
```

SYMBOL TABLE

```
ACK.C = 100000 G    BTMSG2  015021      CP.CMD= 000000 G    C$SVEC= 000037      EXSUB   007240 G
ADR   = 000020 G    BTMSG3  015076      CP.CNT= 000006 G    C$TPRI= 000013      E$END = 002100
ALLEOT  003531 G    BTPT    003516 G    CRLF    005275 G    DATARD  003416 G    E$LOAD= 000035
ASSEMB= 000010      BTRPT   020446      CRLFSP  005300 G    DATAWT  003414 G    FATSM   004455 G
ATTN?   004417 G    BTO     003000 G    CTCC    003456 G    DATCNT= 010000 G    FIRSTU  017300 G
AUDRPM  004727 G    BT1     003052 G    CVC.C = 040000 G    DATRAT  003412 G    FMT.CO= 000040 G
AUTODM  023453      BT2     003124 G    C$AU  = 000052      DEVTBL  002536 G    FMT.C1= 000100 G
BADTM   030735      BT3     003176 G    C$AUTO= 000061      DFPTBL  002176 G    FTLCNT  003320 G
BADTSW  002211 G    BYTECO= 000403      C$BRK = 000022      DFTSCH= 000040 G    FUNRM   004435 G
BFSEQ   024616      CHAR    002220 G    C$BSEG= 000004      DIA   = 100006 G    F$AU  = 000015
BFSEQ0  024642      CHARM   031141      C$BSUB= 000002      DIABLK= 003414 G    F$AUTO= 000020
BFSEQ1  024714      CHGFLG  002217 G    C$CEFG= 000045      DIACNT= 000020 G    F$BGN = 000040
BFSEQ2  024726      CHGM    031122      C$CLCK= 000062      DIAGMC= 000000      F$CLEA= 000007
BFSEQ3  025020      CHKERR  012160 G    C$CLEA= 000012      DINT    002212 G    F$DU  = 000016
BFSEQ4  025072      CH.EAI= 000040 G    C$CLOS= 000035      DINTM   030762      F$END = 000041
BFSEQ5  025134      CH.ERI= 000020 G    C$CLP1= 000006      DLY   = 000020 G    F$HARD= 000004
BFSEQ6  025206      CH.ESS= 000200 G    C$CVEC= 000036      DLY.C = 000020 G    F$HW  = 000013
BFSEQ7  025240      CKDATA  016664 G    C$DCLN= 000044      DRI   = 100013 G    F$INIT= 000006
BFSEQ8  025272      CKDCNT  017274      C$DODU= 000051      DROPDM  004700 G    F$JMP = 000050
BFSEQ9  025324      CKDFF   017276      C$DRPT= 000024      DROPED  003527 G    F$MOD = 000000
BFSE10  025346      CKHAE   017706 G    C$DU  = 000053      DROPN   017622      F$MSG = 000011
BGNFLG= 003464      CKHRTN  017774      C$EDIT= 000003      DROPU   017402 G    F$PROT= 000021
BINC    015722      CLN   = 101012 G    C$ERDF= 000055      DROPUA  017532      F$PWR = 000017
BIT0  = 000001 G    CLRERR  011656 G    C$ERHR= 000056      DRORTN  017610      F$RPT = 000012
BIT00 = 000001 G    CLRFLG  002204 G    C$ERRO= 000060      DTAERM  005460 G    F$SEG = 000003
BIT01 = 000002 G    CLRM    030600      C$ERSF= 000054      DTAER2  004761 G    F$SOFT= 000005
BIT02 = 000004 G    CMDAC   007600 G    C$ERSO= 000057      DTAER3  005030 G    F$SRV = 000010
BIT03 = 000010 G    CMDASC  003732 G    C$ESCA= 000010      DTAER4  005072 G    F$SUB = 000002
BIT04 = 000020 G    CMDD    002222 G    C$ESEG= 000005      DTAER5  005113 G    F$SW  = 000014
BIT05 = 000040 G    CMDLG   003434 G    C$ESUB= 000003      EF.CON= 000036 G    F$TEST= 000001
BIT06 = 000100 G    CMDPKM  004164 G    C$ETST= 000001      EF.NEW= 000035 G    GCMDA   007654 G
BIT07 = 000200 G    CMDPKT  002314 G    C$EXIT= 000032      EF.PWR= 000034 G    GENPAT  010264 G
BIT08 = 000400 G    CMDSAV  003430 G    C$GETB= 000026      EF.RES= 000037 G    GES   = 100017 G
BIT09 = 001000 G    CMDSEQ  003542 G    C$GETW= 000027      EF.STA= 000040 G    GETSTM  005241 G
BIT1  = 000002 G    CMDSE2  003552 G    C$GMAN= 000043      EINC    015730      GIT     010642
BIT10 = 002000 G    CMDTBL  003644 G    C$GPHR= 000042      END   = 177777 G    GOWAIT  011274 G
BIT11 = 004000 G    CMDWRD  003426 G    C$GPLO= 000030      ENDERF= 003476      GSCPK   002324 G
BIT12 = 010000 G    CMD.C0= 000001 G    C$GPRI= 000040      ENDFLG= 003534      G$CNTO= 000200
BIT13 = 020000 G    CMD.C1= 000002 G    C$INIT= 000011      ENDSP   030576      G$DELM= 000372
BIT14 = 040000 G    CMD.C2= 000004 G    C$INLP= 000020      ENDSP1  030404      G$DISP= 000003
BIT15 = 100000 G    CMD.C3= 000010 G    C$MANI= 000050      ENDSP2  030406      G$EXCP= 000400
BIT2  = 000004 G    CMD.C4= 000020 G    C$MEM = 000031      EOTFLG  003506 G    G$HILI= 000002
BIT3  = 000010 G    CMD2M   031166      C$MSG = 000023      ERCVER  002207 G    G$LOLI= 000001
BIT4  = 000020 G    CMD3M   031240      C$OPEN= 000034      ERLOG   003472 G    G$NO  = 000000
BIT5  = 000040 G    CMD4M   031246      C$PNTB= 000014      ERRREC  003475 G    G$OFFS= 000400
BIT6  = 000100 G    CMD5M   031254      C$PNTF= 000017      ERS   = 100411 G    G$OFSI= 000376
BIT7  = 000200 G    CMD6M   031262      C$PNTS= 000016      ERSFLG  003533 G    G$PRMA= 000001
BIT8  = 000400 G    CMD7M   031270      C$PNTX= 000015      EVL   = 000004 G    G$PRMD= 000002
BIT9  = 001000 G    CMD8M   031276      C$QIO = 000377      EWMSG   012526      G$PRML= 000000
BOF   = 000400 G    CMPDAT  003410 G    C$RDBU= 000007      EWPRNT  002216 G    G$RADA= 000140
BORERS  015146 G    CNTBGN= 002560      C$REFG= 000047      EWPT    031064      G$RADB= 000000
BPCRM   031174      CNTEND= 003330      C$RESE= 000033      EWRTRY  012632 G    G$RADD= 000040
BRCPK   002330 G    CNTLEN= 000550 G    C$REVI= 000003      EXALL   006744 G    G$RADL= 000120
BRFCNT  003424 G    CODELM  004054 G    C$RFLA= 000021      EXARTN  007236      G$RADO= 000020
BRF.C = 004000 G    COUNTE= 050416      C$RPT = 000025      EXCRTN  011272      G$XFER= 000004
BTADDR  002550 G    CP.ADH= 000004 G    C$SEFG= 000046      EXCUTE  010654 G    G$YES = 000010
BTMSG1  014734      CP.ADL= 000002 G    C$SPRI= 000041      EXPBOT  003520 G    HAE     002206 G
```

SYMBOL TABLE

```
HAEM     030646
HALTM    004124 G
HELP   = 000000
HERE     003340
HOE    = 100000 G
HRDCNT   003310 G
IBE    = 010000 G
IDV    = 000040 G
IER    = 020000 G
IE.C   = 000200 G
INIT10   021664
INIT15   022104
INIT16   022124
INTFLG   003476 G
INTPRI = 000340 G
IRE      003526 G
IREC     002210 G
IRECM    030714
IREM     031005
ISR    = 000100 G
IXE    = 004000 G
I$AU   = 000041
I$AUTO = 000041
I$CLN  = 000041
I$DU   = 000041
I$HRD  = 000041
I$INIT = 000041
I$MOD  = 000041
I$MSG  = 000041
I$PROT = 000040
I$PTAB = 000041
I$PWR  = 000041
I$RPT  = 000041
I$SEG  = 000041
I$SETU = 000041
I$SFT  = 000041
I$SRV  = 000041
I$SUB  = 000041
I$TST  = 000041
JLOC     003452 G
JLOOP    003450 G
JMP    = 000040 G
JMPMSG   031036
JMPMS2   031256
JMP.C  = 000040 G
J$JMP  = 000167
LENMSK   003436 G
LOE    = 040000 G
LOG      015436 G
LOOPCT   003525 G
LOT    = 000010 G
L$ACP    002110 G
L$APT    002036 G
L$AU     023710 G
L$AUI    002070 G
L$AUTO   023160 G
L$CCP    002106 G
L$CLEA   023602 G
L$CO     002032 G
L$DEPO   002011 G
L$DESC   002140 G
L$DESP   002076 G
L$DEVP   002060 G
L$DISP   002124 G
L$DLY    002116 G
L$DTP    002040 G
L$DTYP   002034 G
L$DU     023644 G
L$DUT    002072 G
L$DVTY   002166 G
L$EF     002052 G
L$ENVI   002044 G
L$ETP    002102 G
L$EXP1   002046 G
L$EXP4   002064 G
L$EXP5   002066 G
L$HARD   027750 G
L$HIME   002120 G
L$HPCP   002016 G
L$HPTP   002022 G
L$HW     002176 G
L$ICP    002104 G
L$INIT   021664 G
L$LADP   002026 G
L$LAST   031510 G
L$LOAD   002100 G
L$LUN    002074 G
L$MREV   002050 G
L$NAME   002000 G
L$PRIO   002042 G
L$PROT   021656 G
L$PRT    002112 G
L$REPP   002062 G
L$REV    002010 G
L$RPT    020052 G
L$SOFT   030022 G
L$SPC    002056 G
L$SPCP   002020 G
L$SPTP   002024 G
L$STA    002030 G
L$SW     002204 G
L$TEST   002114 G
L$TIML   002014 G
L$UNIT   002012 G
L10000   002202
L10001   002312
L10002   005624
L10003   006550
L10004   006556
L10005   006564
L10006   006572
L10007   006600
L10010   021654
L10012   023156
L10013   023446
L10014   023642
L10015   023706
L10016   024002
L10017   025370
L10020   024136
L10021   024162
L10022   024202
L10023   024222
L10024   024242
L10025   024262
L10026   024302
L10027   024322
L10030   024342
L10031   024362
L10032   024420
L10033   024604
L10034   026044
L10035   026522
L10036   026666
L10037   027000
L10040   027744
L10041   030020
L10042   031304
L10043   031514
L10045   031520
MBR    = 100012 G
MEMOM    023054
MISCFG   003537 G
MOD.C0 = 000400 G
MOD.C1 = 001000 G
MOD.C2 = 002000 G
MOD.C3 = 004000 G
MOVMSG   011726 G
MSGCNT = 000016 G
MSGPKA   002506 G
MSGPKT   002340 G
MSGPK0   002356 G
MSGPK1   002374 G
MSGPK2   002412 G
MSGPK3   002430 G
MS.RFC = 000004 G
MS.XS0 = 000006 G
MS.XS1 = 000010 G
MS.XS2 = 000012 G
MS.XS3 = 000014 G
NCMD.C = 177740 G
NCNT     003420 G
NCNT1    003422 G
NEXTSP   030070
NEXTU    017346 G
NINUSE = 177774 G
NOINTM   004503 G
NRDYM    023544
NSSRM    004353 G
NUMBM    031206
NURTY1   005155 G
OFLINM   005211 G
ONEFIL = 000001
OPFLAG   003540 G
OPP.C  = 020000 G
O$APTS = 000000
O$AU   = 000001
O$BGNS = 000001
O$BGNS = 000001
O$DU   = 000001
O$ERRT = 000000
O$GNSW = 000001
O$POIN = 000001
O$SETU = 000001
PASCNT   003260 G
PATCH    031304 G
PATERN   003454 G
PATR0    010350 G
PATR1    010406 G
PATR2    010426 G
PATR3    010436 G
PATR4    010462 G
PATR5    010474 G
PATR6    010506 G
PATR7    010526 G
PATR8    010652 G
PATTBL   010326
PATTM    031226
PCMDWD   003432 G
PIRE     002213 G
PNT    = 001000 G
PRI    = 002000 G
PRI00  = 000000 G
PRI01  = 000040 G
PRI02  = 000100 G
PRI03  = 000140 G
PRI04  = 000200 G
PRI05  = 000240 G
PRI06  = 000300 G
PRI07  = 000340 G
PRXST    017624 G
PTCMDS   027616
PWRFLG   003535 G
RAMDAT   003346 G
RAMDUM   014050 G
RAMER    015736 G
RAMFHR   005306
RAMHLD   003342
RAMIOP   005365
RAMLIN   005450
RAMM     031040
RAMPD    005436
RAMRSH   003344
RAMSIZ   003406 G
RAMWRT   002214 G
RANB     003440 G
RANBC  = 155624 G
RANCMD   025652
RANDOM   003521 G
RANP   = 000007 G
RANRD    025712
RANS     003442 G
RANSC  = 032561 G
RANW     026012
RANWR    025766
RANWV    026000
RAN1     010644 G
RAN2     010646 G
RAN3     010650 G
RCVERM   030672
RDF    = 104001 G
RDR    = 104401 G
RECCNT   003330 G
RECLOG   003471 G
RECRED   006546
RECTAP   006602 G
RECUD    012012 G
REPEAT = 050236
RERM     004632 G
RETRY  = 050234
RETRYC   003464 G
REWRT    015322
RFBC     002660 G
RFCERM   004336 G
RFREC    002760 G
RFUNR    002770 G
RLEXM    004372 G
RNP    = 125401 G
RNOPSC = 177740 G
RNR    = 105401 G
RNYM     004566 G
RPF    = 105001 G
RPR    = 125001 G
RPTCNT   003466 G
RPTFLG   003525 G
RPT1A    020714
RPT1B    020771
RPT1C    021042
RPT1D    021113
RPT1E    021341
RPT1F    021217
RPT1G    021270
RPT1I    021460
RPT1J    021371
RPT1K    021451
RRANV    002205 G
RRBC     002620 G
RRECL  = 000020 G
RRREC    002740 G
RRUNR    002750 G
RRVM     030617
RS1    = 001233 G
RS2    = 007622 G
RS3    = 000000 G
RTLE     014212 G
```

```
RTLRTN  014346          TSBA   = 002456 G      T$$AUT= 010013          WRTYER  003470 G      $ISK0 = 000001
RWCPK   002334 G        TSC.FC= 177717 G       T$$CLE= 010014          WRTYFG  003467 G      $ISK1 = 000001
RWD     = 102010 G      TSC.TC= 177761 G       T$$DAT= 010045          WRUNR   002730 G      $ISK2 = 000001
RWERR   003473 G        TSDB    002456 G       T$$DU = 010015          WSSR    011672 G      $ISK3 = 000001
R5SAVE  003460 G        TSSR    002466 G       T$$HAR= 010041          WTM     = 100011 G    $ISK4 = 000001
SCCNT   003270 G        TSSREG  003462 G       T$$HW = 010000          WTR     = 101011 G    $ISK5 = 000001
SCERM   004312 G        TSVCT   002476 G       T$$INI= 010012          WTV     = 104105 G    $ISK6 = 000001
SCH     = 140004 G      TS.A16= 000400 G       T$$MSG= 010003          WTVERM  004246 G      $LOCTA= 177777
SCHBK   002446 G        TS.A17= 001000 G       T$$PC = 000001          WTYBRF  014732        $LSTIN= 000001
SCHCNT= 000010 G        TS.NBA= 002000 G       T$$PRO= 010011          WTYCMD  014726        $LSTTA= 000001
SEQEND  003632 G        TS.NXM= 004000 G       T$$PTA= 010044          WTYWRD  014730        $LVTAG= 050416
SETCH   006700 G        TS.OFL= 000100 G       T$$RPT= 010010          X$ALWA= 000000        $NESTL= 177777
SETRW   006724 G        TS.RMR= 010000 G       T$$SOF= 010042          X$FALS= 000040        $NSK0 = 000120
SETUP   007706 G        TS.SC = 100000 G       T$$SRV= 010007          X$OFFS= 000400        $NSK1 = 000120
SFF     = 105010 G      TS.SPE= 020000 G       T$$SUB= 010033          X$TRUE= 000020        $NSK2 = 000110
SFPTBL  002204 G        TS.SSR= 000200 G       T$$SW = 010001          XO.BOT= 000002 G      $NSK3 = 000110
SFR     = 105410 G      TS.UPE= 040000 G       T$$TES= 010040          XO.EOT= 000001 G      $NSK4 = 000110
SRF     = 104010 G      TS4AUR  027774         T1      024004 G        XO.LET= 020000 G      $NSK5 = 000110
SRR     = 104410 G      TS4CL   002526 G       T1SWB   003530 G        XO.ONL= 000100 G      $NSK6 = 000110
STAERM  005626 G        TS4INT  002516 G       T1.1    024016          XO.RLL= 010000 G      $SAVE = 000001
STAER1  006140          TS4INO  006552 G       T1.10   024344          XO.RLS= 040000 G      $SAVLE= 177777
STAER2  006316          TS4IN1  006560 G       T1.11   024364          XO.TMK= 100000 G      $SELLE= 177777
STAER3  006375          TS4IN2  006566 G       T1.12   024450          X1.EWN= 000010 G      $SSK0 = 050473
STAER4  006433          TS4IN3  006574 G       T1.2    024140          X2.OPM= 100000 G      $TAGLE= 177777
STAER5  006453          TS4VCT  030011         T1.3    024164          X3.DCK= 000010 G      $TAGNU= 050511
STAER6  006262          T$ARGC= 000001         T1.4    024204          X3.RNY= 157400 G      $TEMP = 000402
STAER7  006232          T$CODE= 023004         T1.5    024224          ZROPAT  010412        $TSK0 = 050472
STAFLG  003534 G        T$ERRN= 000001         T1.6    024244          $BGNLE= 000002        $TSK1 = 050473
STREAM  003532 G        T$EXCP= 000000         T1.7    024264          $BRJMP= 177777        $TSK2 = 050474
SVCGBL= 000000          T$FLAG= 000041         T1.8    024304          $BSK0 = 050234        $TSK3 = 050475
SVCINS= 000001          T$FREE= 031520         T1.9    024324          $BSK1 = 050236        $TSK4 = 050510
SVCSUB= 000000          T$GMAN= 000000         T2      025372 G        $BSK2 = 050416        $TSK5 = 050507
SVCTAG= 000000          T$HILI= 000010         T3      026046 G        $ERFLG= 000400        $TSK6 = 050503
SVCTST= 000000          T$LAST= 000001         T4      026524 G        $F$AND= 000310        $TSK7 = 050505
SWBFLG  003524 G        T$LOLI= 000000         T5      026670 G        $F$BAD= 000401        $U    = 000403
SWB.C = 010000 G        T$LSYM= 010000         T5WEOT  027640          $F$BLA= 000170        $$ARGC= 000000
SYMD  = 000007          T$LTNO= 000006         T6      027002 G        $F$CAS= 000150        $$BYTE= 000403
SYMS  = 000007          T$NEST= 177777         UAM     = 000200 G      $F$DEC= 000220        $$CASE= 000000
S$LSYM= 010000          T$NS0 = 000000         UNL     = 100412 G      $F$DO = 000340        $$DST = 000000
TAPCAP  021571          T$NS1 = 000005         UNREC   003474 G        $F$FAL= 000405        $$ELOC= 000402
TCCRA   012612          T$NS2 = 000002         URERM   004654 G        $F$GOO= 000400        $$ERFL= 000000
TCC0    012740 G        T$PCNT= 000000         VFEXC   016254 G        $F$IF = 000110        $$FLAG= 000001
TCC1    012762 G        T$PTAB= 010044         VFISU   016540 G        $F$INC= 000210        $$FROM= 000000
TCC2    013000 G        T$PTHV= 000001         VFYCNT  003300 G        $F$LOO= 000200        $$INH = 000403
TCC3    013146 G        T$PTNU= 000001         VFYDAT  016142 G        $F$NAM= 000160        $$LOC = 027155
TCC4    013170 G        T$SAVL= 177777         VFYFLG  003522 G        $F$NO = 000403        $$LOCN= 000000
TCC5    013670 G        T$SEGL= 177777         VFY.C = 000100 G        $F$OR = 000320        $$REG = 177777
TCC6    013772 G        T$SIZE= 000004         WLKZRO  010442          $F$RTI= 000350        $$RETU= 000000
TCC7    014026 G        T$SUBN= 000000         WRBC    002560 G        $F$RTN= 000300        $$RTN1= 000000
TCCRTN  013144          T$TAGL= 177777         WRECL = 000020 G        $F$SEL= 000140        $$RTN2= 000000
TIME1   003444 G        T$TAGN= 010046         WRITEM  017776 G        $F$THE= 000330        $$SRC = 000000
TIME2   003446 G        T$TEMP= 000000         WRR     = 105005 G      $F$TRU= 000404        $$TGSV= 000000
TOERM   004271 G        T$TEST= 000006         WRREC   002720 G        $F$UNT= 000130        $$TGS1= 000000
TOOMM   004542 G        T$TSTM= 177777         WRT     = 104005 G      $F$WHI= 000120        $$TGS2= 000000
TRAP04  003536 G        T$TSTS= 000001         WRTY    014350 G        $F$YES= 000402        $$TO  = 000000
TRAP4   023574 G        T$$AU = 010016         WRTYCT  003250 G        $IFLEV= 177777        $$$TAG= 050000
TSAM    004520 G
```

```
. ABS.  031520     000
        000000     001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  62895 WORDS  ( 246 PAGES)
DYNAMIC MEMORY:  19748 WORDS  ( 75 PAGES)
ELAPSED TIME:  00:38:39
CZTKIA,BIN,CZTKIA/-SP/CR=SVC/ML,SPMACJ/ML,CZTKIA
```