TS11 TS11 DATA RELIAB
CZTSHD0

AH-E455D-MC
FICHE 01 OF 01

JUL 1984
digital
Made In USA

```
                .REM    \

                        IDENTIFICATION

PRODUCT CODE:   AC E454D MC

PRODUCT NAME:   CZTSHDO T511 DATA RELIAB

PRODUCT DATE:   15 MARCH 1984

MAINTAINER:     DIAGNOSTIC ENGINEERING

AUTHOR:         J. HITT
```

```
                                        USER DOCUMENTATION

                                        USER DOCUMENTATION TABLE OF CONTENTS


                                        GLOSSARY

                                1.0     GENERAL INFORMATION

                                        1.1     PROGRAM ABSTRACT

                                                1.1.1   FUNCTIONAL DESCRIPTION
                                                1.1.2   STRUCTURE OF PROGRAM
                                                1.1.3   MEMORY MAP
                                                1.1.4   DIAGNOSTIC INFORMATION
                                                        1.1.4.1 SCOPE
                                                        1.1.4.2 ERROR RECOVERY
                                                        1.1.4.3 WRITE ERROR RECOVERY
                                                                1.1.4.3.1       MEDIA/OPERATIONAL
                                                                                SELECTIVE WRITE-ERROR RE
                                                                1.1.4.3.2       OPERATIONAL WRITE ERROR
                                                        1.1.4.4 DIAGNOSTIC TIMING ADJUSTMENT

                                        1.2     SYSTEM REQUIREMENTS

                                                1.2.1   HARDWARE REQUIREMENTS
                                                1.2.2   SOFTWARE REQUIREMENTS

                                        1.3     RELATED DOCUMENTS AND STANDARDS

                                        1.4     DIAGNOSTIC HIERARCHY PREREQUISITES

                                        1.5     ASSUMPTIONS

                                        1.6     DIAGNOSTIC HISTORY

                                2.0     OPERATING INSTRUCTIONS


                                        2.1     HARDWARE PARAMETERS

                                        2.2     SOFTWARE PARAMETERS

                                                2.2.1   TS04 COMMAND LIST
                                                2.2.2   DATA PATTERNS
```

```
105                                                 3.0     ERROR INFORMATION
106
107                                                         3.1     ERROR REPORTING
108
109                                                                 3.1.1   ERROR #1    COMMAND PACKET ADDRESS IS NOT ON A M
110                                                                 3.1.2   ERROR #2    TS04 NOT READY
111                                                                 3.1.3   ERROR #3    NO RESPONSE ERRORS
112                                                                 3.1.4   ERROR #4    NO INTERRUPT ERROR
113                                                                 3.1.5   SPECIAL CONDITION ERRORS
114
115                                                                         3.1.5.1  ERROR #5   TCC0, UNDEFINED SPECIAL COND
116                                                                         3.1.5.2  ERROR #6   TCC1, ATTENTION CONDITION
117                                                                         3.1.5.3  ERROR #7   TCC2, TAPE STATUS ALERT
118                                                                         3.1.5.4  ERROR #8   TCC3, FUNCTION REJECT
119                                                                         3.1.5.5  ERROR #9 - TCC4, RECOVERABLE ERROR
120                                                                         3.1.5.6  ERROR #10  TCC5, RECOVERABLE ERROR
121                                                                         3.1.5.7  ERROR #11  TCC6, UNRECOVERABLE ERROR
122                                                                         3.1.5.8  ERROR #12- TCC7, FATAL SUBSYSTEM ERROR
123
124                                                                 3.1.6   ERROR #13   RFC NON ZERO ERROR
125                                                                 3.1.7   ERROR #14 - RETRY LIMIT EXCEEDED
126                                                                 3.1.8   ERROR #15   TOO MANY INTERRUPTS
127                                                                 3.1.9   ERROR #16   CAPSTAN RUNAWAY
128                                                                 3.1.10  ERROR #17   DATA COMPARE ERRORS
129
130                                                         3.2     ERROR HALTS
131
132
133                                                 4.0     PERFORMANCE REPORT
134
135
136                                                 5.0     TEST SUMMARIES
137
138                                                         5.1     TEST 1   BASIC FUNCTIONS
139                                                         5.2     TEST 2   DATA RELIABILITY
140                                                         5.3     TEST 3   WRITE COMPATABILITY/WRITE UTILITY
141                                                         5.4     TEST 4   READ COMPATABILITY/READ UTILITY
142                                                         5.5     TEST 5   EXECUTE OPERATOR SELECTED COMMAND SEQUENCE
143
144
```

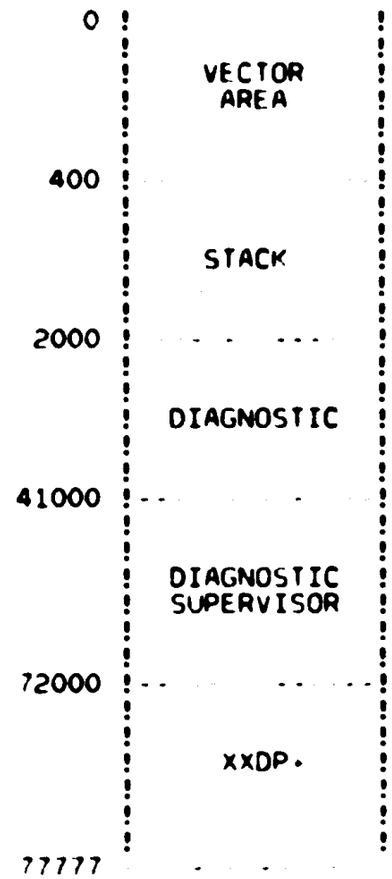
GLOSSARY

ACT	AUTOMATED COMPUTER TEST SYSTEM
APT	AUTOMATED PRODUCT TEST SYSTEM
BYTE/RECORD/FILE COUNT BRF	IS STORED IN THE 4TH WORD OF THE COMMAND PACKET AND IT'S USE BY THE TS04 DEPENDS ON THE TYPE OF COMMAND.
CMD	TS04 COMMAND (SEE 2.3.14.1 FOR LIST OF COMMANDS)
COMMAND PACKET CMDPK	FOUR WORD PACKET IN THE CPU MEMORY WHICH CONTAINS ALL INFORMATION NEEDED BY THE TS04 TO EXECUTE A COMMAND.
EXTENDED STATUS	FOUR WORDS OF TS04 STATUS WHICH ARE TRANSFERRED AS PART OF THE MESSAGE PACKET AT THE COMPLETION OF A COMMAND.
MESSAGE PACKET	SEVEN WORD PACKET IN THE CPU MEMORY INTO WHICH THE TS04 STORES STATUS AT THE COMPLETION OF A COMMAND.
PC	PROGRAM COUNTER
PSW	PROCESSOR STATUS WORD
RESIDUAL FRAME COUNT RFC	THIS COUNT IS PART OF THE MESSAGE PACKET AND CONTAINS THE NUMBER OF BYTES/RECORDS /FILES REMAINING TO BE PROCESSED AT THE COMPLETION OF A COMMAND.
SPECIAL CONDITION SPEC COND	TSS4 BIT1[illegible] WHEN SET, INDICATES THAT THE LAST COMMAND DID NOT COMPLETE WITH-OUT INCIDENT.
TERMINATION CLASS CODE	THREE BIT CODE IN THE TSSR WHICH INDI

```
                        TCC             CATES THE TYPE OF COMMAND TERMINATION.

                        TSBA            TAPE SYSTEM BUS ADDRESS REGISTER.

                        TSDB            TAPE SYSTEM DATA BUFFER REGISTER.

                        TSSR            TAPE SYSTEM STATUS REGISTER.

                        XST0            EXTENDED STATUS REGISTER 0

                        XST1            EXTENDED STATUS REGISTER 1

                        XST2            EXTENDED STATUS REGISTER 2

                        XST3            EXTENDED STATUS REGISTER 3

                        XXDP.           XXDP. IS A 'CATCH ALL" NAME FOR A GROUP OF PDP 1
                                        DIAGNOSTIC PACKAGES AVAILABLE ON MULTIMEDIA.
```

```
214
215                                 1.0 GENERAL INFORMATION
216                                     - -
217
218                                 1.1 PROGRAM ABSTRACT
219                                     --- -   - -
220
221
222                                 1.1.1 FUNCTIONAL DESCRIPTION
223                                           -   -----
224
225                                       THIS PROGRAM CAN BE USED AS A BASIC FUNCTION TEST, A DATA RELIABILITY
226                                       TEST, A COMPATABILITY TEST, OR TO EXECUTE A SEQUENCE OF OPERATOR
227                                       SELECTED COMMANDS.
228
229                                 1.1.2 STRUCTURE OF PROGRAM
230                                       --- - -  ----------
231
232                                       THIS DIAGNOSTIC IS A SINGLE PROGRAM
233                                       FROM THE STANDPOINT OF THE  DIAGNOSTIC USER, BUT IT CONTAINS A CONTROL
234                                       MODULE RELEASED INDEPENDENTLY AS A DIAGNOSTIC SUPERVISOR.
235
236
```

1.1.3 MEMORY MAP



FREE MEMO SPACE FOR WR/RD BFRS OR OTHER PUROSES
IS ALLOCATED BY THE SUPERVISOR ON REQUEST OR CHOOSEN
BY PROGRAMMER TO RESIDE BETWEEN THE DIAG AND THE
SUPERVISOR.

1.1.4 DIAGNOSTIC INFORMATION

1.1.4.1 SCOPE

THIS DIAGNOSTIC CAN TEST UP TO 4 UNITS SIMULTANEOUSLY. THE 4 UNITS ARE ASSIGNED LOGICAL UNIT NUMBERS 0 3 BY THE DIAGNOSTIC.

THERE ARE 5 TESTS IN THIS PROGRAM:

- TEST 1 BASIC FUNCTIONS.
- TEST 2 DATA RELIABILITY.
- TEST 3 - WRITE COMPATABILITY/WRITE UTILITY.
- TEST 4 - READ COMPATABILITY/READ UTILITY.
- TEST 5 - 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 RECOVER PROCEDURES, SEE SECTION 3.0 (ERROR REPORTING).

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 ENTERED UPON DETECTIION OF A WRITE RECOVERABLE ERROR.
THE WRITE RETRY SUBROUTINE REWRITES RECORD IN SAME SPOT ON TAPE:
REPEAT 4 TIMES.
IF ALL 4 REPEATS ARE GOOD, RECORD IS CONSIDERED AS RECOVERED AND A RECOVERABLE WRITE ERROR IS LOGGED AT THAT RECORD NUMBER.
IF ANY OF THE 4 REPEAT FAILS, ERASE BAD RECORD, LOGG SUSPECTED

```
                        BAD SPOT AT THAT RECORD NUMBER, RETRY AGAIN 3 INCHES FURTHER DOWN TAPE.
                        RETRY 4 TIMES, UP TO 4 REPEATS EACH.
                        IF RECORD CANNOT BE WRITTEN WITHOUT RECOVERABLE ERROR  AFTER 4 RETRIES,
                        ERASE RECORD, REPORT RETRY FAILED ON BAD SPOT.
                        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 .

                        TWENTY (20) BAD SPOTS MAXIMUM ARE ALLOWED PER TAPE PASS.
                        WHEN 20 BAD SPOTS HAVE BEEN LOGGED, ON SAME RECORD NUMBER OR NOT,
                        TAPE IS CONSIDERED DEFECTIVE:  A BAD TAPE OVERFLOW MESSAGE IS PRINTED
                        AND UNIT IS REWOUND, THEN DROPPED.

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

                        ALL BADLY WRITTEN RECORDS FLAGGED WITH RECOVERABLE ERRORS ARE ERASED
                        UNTIL RECOVERED, INCLUDING THE RECORD AT THE 20TH BAD SPOT,
                        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 GRO
                        UP TO 20 FEET OF ERASE GAP COULD RESULT WHEN RETRYING TO RECOVER
                        A SINGLE RECORD,IF NO BAD SPOT WERE PREVIOUSLY DETECTED.
                        THAT LONG STRETCH OF  BAD TAPE WOULD THEN BE FLAGGED WITH 20
                        BAD SPOTS AT SAME RECORD NUMBER AND THE TAPE CONSIDERED DEFECTIVE.

                        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
                        THE BAD SPOTS ACTUALLY PRECEDES 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 RETR

                        THE STATISTICAL REPORT PRINTED AT THE END OF TEST 2 OR UPON A 'PRINT' RE
                        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 REQUE
                        ON ALL UNITS SELECTED. THOSE 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, MOST OF THE TIME.

                        TO CLEAR CUMULATIVE COUNTS, ANSWER 'Y' TO:  CLEAR COUNTERS (L) Y ?.
                        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 ARE KNOWN 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 FORM TAPE PASS TO TAPE PASS
                        FOR THE BAD SPOTS ID TO REMAIN CONSISTENT IN THOSE TAPE PASSES.

                        EXAMPLE OF A TAPE PASS PRINTS:

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


                        CZTSH SFT ERR  00009 ON UNIT 00 TST 002 SUB 000 PC: 012100
                        RECOVERABLE ERROR
                        WRT CMD FAILED   UNIT 0   PASS:    1   RECORD:10210
                        PREVIOUS CMD WAS  WRT
                        CMDPKT  TSBA    RFC     TSSR   TCC
                        100205  002406  000000  100210  4
                        026600
                        000000
                        004000
                        XST0    XST1    XST2    XST3
                        000350  000002  100010  000000
                        RECOVERED ON RETRY # 1
                        ↑C
                        DR>PRI

                        UNIT 0   PASS:    1   RECORD:10210
                        BYTES WRITTEN    0,272,279,691
                        BYTES READ REV   0,301,123,654
                        BYTES READ REV   0,301,120,381
                                              WRT    RDR    RDF
                        RECOVERABLE ERRORS     1      0      0
                        UNRECOVERABLE ERRORS   0      0      0
                        WRITE RETRIES          3

                         2 BAD SPOTS THIS TAPE PASS PRECEDING RECORD #:
```

```
                                        6     6
                                SPEC COND   HARD   FATAL   COMPARE
                                      2      0       0       0
                                DR>

                                THIS EXAMPLE SHOWS:
                                        RECORD 6 RECOVERED ON 2ND RETRY GROUP
                                        THE 2 BAD SPOTS RESIDE IN A 18 INCH ERASE GAP BETWEEN RECORDS 5
                                        RECORD 10210 RECOVERED ON 1ST RETRY OF 4 GOOD REPEATS
                                        3 WRITE GROUP RETRIES ATTEMPTED, RESULTING IN:
                                                1 RECOVERABLE  WRT ERR FROM RECORD 10210
                                                2 BAD SPOTS BETWEEN RECORDS 5 AND 6


                        1.1.4.3.2       OPERATIONAL WRITE ERROR RECOVERY ALGORITHM

                                WHEN THIS ALGORITHM IS SELECTED, THE TS11 WRITE RETRY COMMAND
                                IS ISSUED UP TO 16 TIMES OR UNTIL RECORD IS RECOVERED, ON
                                A WRITE RECOVERABLE ERROR.  THE WRITE  RETRY COMMAND CONSISTS
                                OF A SPACE REVERSE OVER THE BAD RECORD, THEN AN ERASE OF 3 INCHES
                                OF TAPE AND REWRITE OF THE RECORD.  THAT COMPOSITE COMMAND
                                DOES NOT ALLOW TO DETECT BAD SPOTS ON TAPE.
                                THEREFORE NO BAD TAPE SPOTS STATUS IS PRINTED.

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


                        1.1.4.4 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
                                COMPLETION.  THESE DELAYS ARE NOT CALIBRATED AND SIMPLY EXPANDS
                                INTO AN INLINE NESTED LOOP PAIR.  THE COUNT FOR THE OUTER LOOP
                                COMES FROM THE VARIABLE ARGUMENT SUPPLIED BY THE DELAY CALLS.
                                THE COUNT FOR THE INNER LOOP COMES FORM 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 IS EVIDENT, ALL TIMINGS OF THE DIAGNOSTIC MAY BE ADJUSTED
                                TO MATCH MEMORY SPEED AND NOT RESULT IN TIME OUTS, BY PATCHING
                                THAT FIXED DELAY ELEMENT "L$DLY".

                                A PRESET COUNT OF 500 RESIDES AT L$DLY  IN LOCATION 2116 OF THE
                                 HEADER' SECTION.


                1.2 SYSTEM REQUIREMENTS


                        1.2.1  HARDWARE REQUIREMENTS
```

```
                                        PDP 11 PROCESSOR WITH 16K OR MORE OF MEMORY
                                        CONSOLE DEVICE (LA30,LA36,VT50,ETC.)
                                        PROGRAM LOAD DEVICE


                                1.2.2   SOFTWARE REQUIREMENTS


                                        DIAGNOSTIC SUPERVISOR
```

1.3 RELATED DOCUMENTS AND STANDARDS

XXDP+ USERS MANUAL MD-11-CHQUS
DIAGNOSTIC SUPERVISOR PROGRAM LISTING
PDP-11 DIAGNOSTIC SUPERVISOR INTERFACE SPECIFICATION.
PDP-11 DIAGNOSTIC SUPERVISOR PROGRAMMER'S GUIDE.
TS11/TSO4 PROGRAMMING SPECIFICATION.
TS11/TSO4 ENGINEERING SPECIFICATION.
TS11/TSO4 COMMAND PACKET SPECIFICATION.

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

ORDER OF HOST CPU DIAGNOSTIC USAGE:

1) CONTROL LOGIC 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 WORK PROPERLY. FALSE ERRORS MAY BE REPORTED IF THE PROCESSOR, MEMORY, ETC., DO NOT FUNCTION PROPERLY.

1.6 DIAGNOSTIC HISTORY

REVISION A - OCT 1978
- ORIGINAL RELEASE

REVISION B FEB 1979
- CORRECTED END OF TAPE PROBLEMS IN TESTS 3 5.
- CHANGED DEFAULT VECTOR ADDRESS FROM 150 TO 224.
  DECREASED MAXIMUM RECORD LENGTH FROM 4096 TO 2048 BYTES.

REVISION B - AUG 1979
-DO NOT PRINT RECOVERABLE ERRORS UNLESS REQUESTED BY OPERATOR
 WARN OPERATOR OF UNIT(S) BEING NOT READY OR OFF-LINE.
 DROP UNIT(S) LEFT NOT READY OR OFF-LINE FOR 3.5 MINUTES.
-IMPROVE BEHAVIOR AT EOT
  IN TEST 2, FREEZE UNITS REACHING EOT UNTIL OTHERS
  CATCH-UP INSTEAD OF ALLOWING THEM TO SHUTTLE AT EOT
 -WHEN ALL UNITS REACH EOT, WRITE ONE RECORD BEYOND EOT.
  READ REV THAT EXTRA RECORD TO POSITION TAPE
  SO THAT THE NEXT COMMAND REQUESTED CAN BE EXECUTED.
  THAT EXTRA RECORD SHALL LEAVE A CLEAN IRG GAP AND A VALID
  RECORD TO READ WHEN SHORTER RFAD STOP DISTANCE MIGHT CAUSE
  UNIT TO FLAG EOT ON THAT EXTRA RECORD INSTEAD OF THE
  PREVIOUS ONE. THIS SHOULD ELIMINATE MANY READ ERRORS AT
  EOT AND TAPES RUNNING OFF THE WHEELS.
-WRITE RECORD COUNT ON TAPE.
 PRINT RECORD COUNT READ FROM TAPE IN READ ERROR PRINTS TO
 INDICATE IF POSITION WAS LOST.

• CAUTION •

 INTERPRET THAT 'RECORD READ' COUNT WITH CAUTION.
 IF VERY DIFFERENT FROM RECORD COUNT TRACKED BY THE DIAGNOST
 POSITION IS NOT NECESSARELY LOST. ERRORS IN READING THAT
 RECORD MIGHT HAVE CAUSED RECORD COUNT TO BE ERRONEOUSLY
 READ FROM TAPE.
 IN TEST 2, IF DIAGNOSTIC IS RESTARTED OR CONTINUED, RECORD
 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 DI
 TO GET A PRINT OF THE RECORD COUNT WHEN HALTED.

VERIFY RECORD OF 4000 BYTES INSTEAD OF 22 BYTES.
WHEN COMPARING DATA, CHECK AND PRINT IF NO DATA WAS READ
OR RECORD WAS LONGER THAN EXPECTED.
FREEZE TSSR REG WHEN A COMMAND IS COMPLETED TO AVOID DIFFERE
BETWEEN TSSR AND TCC FETCHED AT DIFFERENT TIMES.
WHEN DROPPING A UNIT, FLAG SECOND PRINT OF EXTENDED STATUS
THE RESULT OF A GET STATUS COMMAND.
WAIT FOR SSR UP BEFORE PRINTING THAT STATUS.
ADJUST "PASS" COUNT OF DIAG TO MATCH 'EOP" PASS COUNT OF SUP
INCREASE NUMBER OF SELECTABLE COMMANDS IN TEST 5 FROM
4 TO 7. DEFAULT COMMAND 6 IS NOW REWIND.
CONVERT DIAG TO REV C OF SUPERVISOR.

```
                                        ADD SEVERAL SECTIONS:
                                          PROTECT TABLE
                                          AUTO-DROP CODE
                                          HARD CODED PARAMETER TABLE
                                REVISION C  OCT 79
                                          ADD MEDIA/OPERATIONAL SELECTIVE WRITE ERROR RECOVERY ALGORI
                                          TO DETECT BAD SPOTS ON TAPE.
                                REVISION D  MARCH 84
                                          - FIX ERROR ROUTINES SO THAT DATA COMPARE ERRORS IN TEST 2
                                          DO NOT CAUSE OTHER PROBLEMS.
```

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 DIAG SHALL RUN ASSUMING 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 ADRRES OF THE TSSR(S). IF 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 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 SC OFTEN TO WARN THE OPERATOR OF UNITS BEING NOT READY OR OFF LINE. THESE UNITS SHALL BE DROPPED AFTER A REASONABLE AMOUNT OF TIME (3 MIN ON A 11/70).

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 ?

PRINT RECOVERABLE ERRORS (L) N ?

HALT AFTER EACH CMD (L) N ?

INHIBIT RECOVERY (L) N ?

BAD TAPE SPOT DETECTION (L) Y ?

DISABLE INTERRUPTS (L) N ?

INHIBIT RFC ERROR REPORTS (L) N ?

CHANGE CMD SEQUENCE (L) N ?

NOTE: THIS QUESTION SHOULD BE ANSWERED (N) UNLESS AN OPERATOR SELECTED SEQUENCE IS TO BE EXECUTED. IF THIS QUESTION WAS ANSWERED (N), NO MORE QUESTIONS WILL BE ASKED. IF THIS QUESTION WAS ANSWERED Y, THE FOLLOWING QUESTIONS MUST BE ANSWERED OR DEFAULTED WITH A <CR> ONLY:

CHARACTERISTICS CODE (O) 40 ?	(0,20,40,200) (OCTAL)
CMD/2 (D) 13 ?	(1 27) (DECIMAL)
BRF COUNT (D) 1 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 1 ?	(1 32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)
CMD/3 (D) 4 ?	(1 27) (DECIMAL)
BRF COUNT (D) 2048 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 32000 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)
CMD/4 (D) 3 ?	(1-27) (DECIMAL)
BRF COUNT (D) 2048 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 32000 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)
CMD/5 (D) 2 ?	(1-27) (DECIMAL)
BRF COUNT (D) 2048 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 32000 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)
CMD/6 (D) 13. ?	(1-27) (DECIMAL)
BRF COUNT (D) 1 ?	(1 2K) (DECIMAL)
# OF OPERATIONS (D) 1 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)
CMD/7 (D) 27. ?	(1-27) (DECIMAL)
BRF COUNT (D) 2048 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 32000 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)

```
708                                     CMD/8 (D) 27. ?                 (1-27)   (DECIMAL)
709                                     BRF COUNT (D) 2048 ?            (1-2K)   (DECIMAL)
710                                     # OF OPERATIONS (D) 32000 ?     (1-32K)  (DECIMAL)
711                                     PATTERN (D) 7 ?                 (0-8)    (DECIMAL)
712
713                                     NOTE:   THE PROGRAM AUTOMATICALLY INSERTS AN CHARACTERISTIC 40
714                                             AS THE FIRST COMMAND IN THE SEQUENCE TABLE.  IF A
715                                             DIFFERENT CHARACTERISTIC IS DESIRED, THE OPERATOR SHOULD
716                                             ENTER THAT CHARACTERISTIC CODE.  A TOTAL OF 7 COMMANDS
717                                             MAY BE ENTERED IN ADDITION TO THE SET CHARACTERISTICS
718                                             COMMAND.  IF THE OPERATOR WISHES TO USE LESS THAN 7
719                                             COMMANDS, AN END COMMAND MUST BE ENTERED AND THEN A
720                                             CONTROL Z (^Z) CAN BE ENTERED TO TERMINATE SOFTWARE DIA_
```

2.2.1 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. IE. WRITE N RECORDS; READ REVERSE AND CHEC N RECORDS OF DATA; 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 TS11/TSO4 PROGRAMMING SPECIFICATION FOR DESCRIPTION.
24 =	DIA	DIAGNOSTICS. SEE TS11/TSO4 PROGRAMMING SPECIFICATION FOR DESCRIPTION. ODT MUST BE USED TO LOAD DIAGNOSTIC DA INTO THE WRITE BUFFER BEFORE THIS CMD IS ISSUED.
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 # OF OPERATIONS FI
26 =	DLY	DELAY "N' MILISECONDS WHERE N IS DEFINED IN THE # OF OPERATIONS.
27 =	END	END OF COMMAND SEQUENCE.

2.2.2 DATA PATTERN LIST FOR USE IN SOFTWARE DIALOGUE.

PATTERN #	DESCRIPTION.
0	INCREMENTING PATTERN. 0 377.
1	ALL "1"'S PATTERN.
2	ALL "0"'S PATTERN.
3	"1" BIT WALKING FROM R TO L IN A FIELD OF 0" S.
4	"0' BIT WALKING FROM R TO L IF A FIELD OF 1' S.
5	ALTERNATING "1" AND '0' BITS WITH ALTERNATE BYTES COMPL
6	ALTERNATING BYTES OF 000 AND 377.
7	RANDOM DATA PATTERN.
8	NO PATTERN GENERATION.

2.3 EXAMPLES OF SOFTWARE DIALOGUE

2.3.1 BASIC FUNCTION AND DATA RELIABILITY WITH ALL ERROR REPORTING ENABLED

A) RECEIVE PROMPT (DR>)
B) ENTER STA/TES:1-2<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>
PRINT RECOVERABLE ERRORS  (L) N ?        Y<CR>
HALT AFTER EACH CMD (L) N ?              N<CR>
INHIBIT RECOVERY (L) N ?                 N<CR>
BAD TAPE SPOT DETECTION (L) Y ?          Y<CR>
DISABLE INTERRUPTS (L) N ?               N<CR>
INHIBIT RFC ERROR REPORT (L) N ?         N<CR>
CHANGE CMD SEQUENCE (L) N ?              N<CR>
```

```
797                                   2.3.2  TO SET UP A SCOPE LOOP FOR A FAILURE IN BASIC FUNCTIONS.
798
799                                            A)  RECEIVE PROMPT (DR>)
800                                            B)  ENTER STA/TES:1/FLA:LOE:IER:ISR:IDU<CR>
801                                            C)  ANSWER HARDWARE QUESTIONS.
802                                            D)  PROCEED WITH THE FOLLOWING DIALOGUE:
803
804                                                  CHANGE SW (L) ?                          Y<CR>
805                                                  CLEAR COUNTERS (L) N ?                   Y<CR>
806                                                  RESET RANDOM VARIABLES (L) N ?           N<CR>
807                                                  PRINT RECOVERABLE ERRORS  (L) N ?        N<CR>
808                                                  HALT AFTER EACH CMD (L) N ?              N<CR>
809                                                  INHIBIT RECOVERY (L) N ?                 N<CR>
810                                                  BAD TAPE SPOT DETECTION (L) Y ?          N<CR>
811                                                  DISABLE INTERRUPTS (L) N ?               N<CR>
812                                                  INHIBIT RFC ERROR REPORT (L) N ?         Y<CR>
813                                                  CHANGE CMD SEQUENCE (L) N ?              N<CR>
814
815
816                                   2.3.3  TO SET UP A SCOPE LOOP FOR A FAILURE IN DATA RELIABILITY
817
818                                            A)  RECEIVE PROMPT (DR>)
819                                            B)  ENTER STA/TES:5/FLA:IER:ISR:IDU/EOP:1000<CR>
820                                            C)  ANSWER HARDWARE QUESTIONS.
821                                            D)  PROCEED WITH THE FOLLOWING DIALOGUE:
822
823                                                  CHANGE SW (L) ?                          Y<CR>
824                                                  CLEAR COUNTERS (L) N ?                   Y<CR>
825                                                  RESET RANDOM VARIABLES (L) N ?           N<CR>
826                                                  PRINT RECOVERABLE ERRORS  (L) N ?        N<CR>
827                                                  HALT AFTER EACH CMD (L) N ?              N<CR>
828                                                  INHIBIT RECOVERY (L) N ?                 N<CR>
829                                                  BAD TAPE SPOT DETECTION (L) Y ?          N<CR>
830                                                  DISABLE INTERRUPTS (L) N ?               Y<CR>
831                                                  INHIBIT RFC ERROR REPORT (L) N ?         Y<CR>
832                                                  CHANGE CMD SEQUENCE (L) N ?              Y<CR>
833                                                  CHARACTERISTICS CODE (O) 40 ?            40<CR>
834                                                  CMD/2 (D) 5 ?                            13<CR>   (REWIND) (COULD
835                                                  BRF COUNT (D) 2048 ?                      1<CR>
836                                                  # OF OPERATIONS (D) 10 ?                  1<CR>
837                                                  PATTERN (D) 7 ?                           1<CR>
838                                                  CMD/3 (D) 5 ?                             4<CR>   (WRITE) (COULD B
839                                                  BRF (D) 2048 ?                         1000<CR>
840                                                  # OF OPERATIONS (D) 10 ?               10000<CR>
841                                                  PATTERN (D) 7 ?                           1<CR>
842                                                  CMD/4 (D) 5 ?                            27<CR>   (END)   (COULD B
843                                                  BRF (D) 2048 ?                           <^Z>
```

2.4 EXECUTION TIMES

2.4.1 SYSTEM CONFIGURATION

PDP11/34
MOS MEMORY
LA36
TS11/TS04

2.4.2 TEST EXECUTION TIMES

TEST 1 BASIC FUNCTIONS - 30 SECONDS PER PASS.
TEST 2 - DATA RELIABILITY - 45 MINUTES PER PASS.
TEST 3 WRITE COMPATABILITY - 20 MINUTES PER PASS.
TEST 4 - READ COMPATABILITY - 20 MINUTES PER PASS.
TEST 5 - OPERATOR SELECTED SEQUENCE - DEPENDS ON SEQUENCE SELECTED.

NOTE: ALL EXECUTION TIMES ARE SHOWN FOR ONE UNIT OEPRATION.
APPROXIMATELY 10% WILL BE ADDED TO ALL EXECUTION TIMES
FOR EACH ADDITIONAL UNIT.

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, # OF
RECORDS FROM BOT, RECORD READ COUNT, THE COMMAND PACKET, TSSR,
TCC, TSBA, RFC, AND THE EXTENDED STATUS REGISTERS
(SEE 2.3.14.1 FOR LIST OF COMMANDS).

STANDARD ERROR REPORT FORMAT:

```
CZTSH 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 THAT "RECORD READ" COUNT WITH CAUTION.
IF VERY DIFFERENT FROM RECORD COUNT TRACKED BY THE DIAGNOST
POSITION IS NOT NECESSARELY LOST. ERRORS IN READING THAT
RECORD MIGHT HAVE CAUSED RECORD COUNT TO BE ERRONEOUSLY
READ FROM TAPE.
IN TEST 2, IF DIAGNOSTIC IS RESTARTED OR CONTINUED, RECORD
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 DI
TO GET A PRINT OF THE RECORD COUNT WHEN HALTED.

EXAMPLE OF AN ERROR REPORT:

```
CZTSH 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   TS04 NOT READY:

                        BEFORE ANY COMMAND IS ISSUED TO THE TS04, THE SUBSYSTEM READY
                        BIT IN THE TSS4 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 TS04 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 RECEIVED.  THE PROGRAM REPORTS
                        THAT NO INTERRUPT OCCURRED.  THIS IS A FATAL DEVICE ERROR AND
                        THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU
                        OPTION IS USED.


                3.1.5   SPECIAL CONDITION ERRORS:

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

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


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

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


                3.1.5.2 ERROR #6   TERMINATION CLASS CODE 1, ATTENTION CONDITION
```

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

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

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

3.1.5.4 ERROR #8 TERMINATION CLASS CODE 3, FUNCTION REJECT

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

31.5.5 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 RETRY LIMIT IS REACHED BEFORE THE ERROR IS RECOVERED, 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 RETRY LIMIT IS REACHED BEFORE THE ERROR IS RECOVERED, 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. IF DENSITY CHECK IS SET THIS DIAGNOSTIC WILL REWIND AND RETRY THE COMMAND, OTHERWISE 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 INFOR MATION ON THE TYPE OF FATAL ERROR. THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

```
1037
1038
1039                                    3.1.6   ERROR #13   RFC NON ZERO ERROR:
1040
1041                                            IF, AFTER EXECUTION, THE RESIDUAL FRAME COUNT IS NON ZERO, THE
1042                                            ERROR IS REPORTED AND A HARD ERROR IS LOGGED.  THE PROGRAM THEN
1043                                            PROCEEDS NORMALLY.  THE REPORTING AND LOGGING OF THESE ERRORS
1044                                            IS OPTIONAL.
1045
1046
1047                                    3.1.7   ERROR #14   RETRY LIMIT EXCEEDED:
1048
1049                                            ON A WRITE COMMAND THIS IS A FATAL DEVICE ERROR AND THE DEVICE
1050                                            WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.
1051
1052                                            ON A READ COMMAND THIS ERROR IS LOGGED AS A HARD ERROR AND
1053                                            THE PROGRAM PROCEEDS NORMALLY.
1054
1055
1056                                    3.1.8   ERROR #15 - TOO MANY INTERRUPTS:
1057
1058                                            IF MORE THAN ONE INTERRUPT OCCURS PER COMMAND, THIS ERROR IS REPORTED.
1059                                            THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM
1060                                            THE TEST CYCLE UNLESS THE IDU OPTION IS USED.
1061
1062
1063                                    3.1.9   ERROR #16   CAPSTAN RUNAWAY:
1064
1065                                            CAPSTAN DID NOT STOP WITHIN ACCEPTABLE WINDOW AFTER LAST
1066                                            COMMAND.  THE PROGRAM WILL ISSUE A GET STATUS COMMAND BEFORE REPORTING
1067                                            THE ERROR SO THAT THE DEAD TRACK FIELD IN EXTENDED STATUS REGISTER 2
1068                                            WILL CONTAIN THE TACH COUNT WHEN THE TAPE STOPPED.
1069                                            THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM
1070                                            THE TEST CYCLE UNLESS THE IDU OPTION IS USED.
1071
1072
1073                                    3.1.10  ERROR #17   DATA COMPARE ERROR:
1074
1075                                            IF A DATA VALIDATION ERROR OCCURS DURING A WRITE/VERIFY COMMAND,
1076                                            THE PROGRAM PRINTS WHAT THE DATA SHOULD HAVE BEEN AND WHAT THE
1077                                            DATA WAS, AND PRINTS THE BYTE AND RECORD NUMBER THE ERROR OCCURRED
1078                                            ON.  ONLY THE FIRST 10 BYTES IN ERROR PER RECORD ARE PRINTED.
1079                                            THE TOTAL # OF BYTES IN ERROR PER RECORD IS ALSO PRINTED.  A
1080                                            HARD ERROR IS LOGGED AND THE PROGRAM PROCEEDS NORMALLY.
1081
1082
1083                                    3.2 ERROR HALTS
1084
1085
1086                                            ERROR HALTS ARE SUPPORTED PER DESCRIBED IN THE PREVIOUS SECTION
1087                                            WITH /FLAG:HOE.  THERE ARE NO OTHER HALTS.
```

```
1088                                    4.0 PERFORMANCE REPORT
1089                                        - - - - - - - - - -
1090
1091                                        UNIT X   PASS:XXXXX  RECORD:XXXXX
1092                                        BYTES WRITTEN  XXX,XXX,XXX,XXX
1093                                        BYTES READ REV XXX,XXX,XXX,XXX
1094                                        BYTES READ FWD XXX,XXX,XXX,XXX
1095                                                              WRT       RDR      RDF
1096                                        RECOVERABLE ERRORS    XXXXX     XXXXX    XXXXX
1097                                        UNRECOVERABLE ERRORS  XXXXX     XXXXX    XXXXX
1098
1099                                        SPEC COND   HARD   FATAL   COMPARE
1100                                           XXXXX   XXXXX  XXXXX   XXXXX
1101
1102
1103
1104
1105
1106                                    5.0 TEST SUMMARIES
1107                                        - - - - - - -
1108
1109                                    5.1     TEST 1 -        BASIC FUNCTIONS.
1110
1111                                                            EXECUTES AND VERIFIES CORRECT COMPLETION OF ALL TS04 FUN
1112
1113                                                            SUBTEST 1    SET CHAR, DRIVE INIT, GET STATUS.
1114                                                                           • SET CHARACTERISTIC 200.
1115                                                                           • DRIVE INITIATE.
1116                                                                           • SET CHARACTERISTIC  20.
1117                                                                           • GET STATUS
1118                                                                           • SET CHARACTERISTIC  40.
1119                                                                           • PRINT TS04 MICROCODE LEVEL (PASS 1 ONL
1120
1121                                                            SUBTEST 2    REWIND.
1122                                                                           • REWIND.
1123                                                                           • REWIND AT BOT.
1124
1125                                                            SUBTEST 3    WRITE/VERIFY.
1126                                                                           • WRITE/VERIFY PATTERN 1.
1127                                                                           • WRITE/VERIFY PATTERN 2.
1128                                                                           • WRITE/VERIFY PATTERN 3.
1129                                                                           • WRITE/VERIFY PATTERN 4.
1130                                                                           • WRITE/VERIFY PATTERN 5.
1131                                                                           • WRITE/VERIFY PATTERN 6.
1132                                                                           • WRITE/VERIFY PATTERN 0.
1133
1134                                                            SUBTEST 4    WRITE TAPE MARK, ERASE.
1135                                                                           • WRITE TAPE MARK.
1136                                                                           • WRITE 10 RECORDS
1137                                                                           • ERASE 10 TIMES
1138                                                                           • WRITE TAPE MARK.
1139                                                                           • WRITE TAPE MARK RETRY.
1140
1141                                                            SUBTEST 5    SPACE FILES.
1142                                                                           • SPACE 2 FILES REVERSE.
1143                                                                           • SPACE 2 FILES FORWARD.
```

```
1144                                                    • SPACE 2 FILES REVERSE.
1145                                                    • SPACE 2 FILES FORWARD.
1146
1147                                            SUBTEST 6    SPACE RECORDS.
1148                                                    • REWIND.
1149                                                    • SPACE 7 RECORDS FORWARD.
1150                                                    • SPACE 7 RECORDS REVERSE.
1151                                                    • SPACE 7 RECORDS FORWARD.
1152                                                    • SPACE 7 RECORDS REVERSE.
1153
1154                                            SUBTEST 7    WRITE RETRY.
1155                                                    • REWIND.
1156                                                    • WRITE DATA.
1157                                                    • WRITE RETRY.
1158
1159                                            SUBTEST 8    READ REV RETRY.
1160                                                    • READ REVERSE.
1161                                                    • READ NEXT REVERSE.
1162                                                    • READ NEXT FORWARD.
1163
1164                                            SUBTEST 9    READ FWD RETRY.
1165                                                    • READ FORWARD.
1166                                                    • READ PREVIOUS FORWARD.
1167                                                    • READ PREVIOUS REVERSE.
1168
1169                                            SUBTEST 10   CLEAN.
1170                                                    • CLEAN.
1171                                                    • REWIND.
1172
1173                                            SUBTEST 11   - WRITE/VERIFY SWAPPED DATA BYTES.
1174                                                    • WRITE/VERIFY EVEN LENGTH (RECORD 1).
1175                                                    • WRITE/VERIFY ODD LENGTH (RECORD 2).
1176                                                    • SET DATA BYTE SWAP.
1177                                                    • WRITE/VERIFY EVEN LENGTH (RECORD 3).
1178                                                    • WRITE/VERIFY ODD LENGTH (RECORD 4).
1179                                                    • CLEAR DATA BYTE SWAP.
1180
1181                                            SUBTEST 12   READ SWAPPED DATA BYTES.
1182                                                    • READ REV RECORD 4.
1183                                                    • READ REV RECORD 3.
1184                                                    • SET DATA BYTE SWAP.
1185                                                    • READ REV RECORD 2.
1186                                                    • READ REV RECORD 1.
1187                                                    • READ FWD RECORD 1.
1188                                                    • READ FWD RECORD 2.
1189                                                    • CLEAR DATA BYTE SWAP.
1190                                                    • READ FWD RECORD 3.
1191                                                    • READ FWD RECORD 4.
```

```
1192                                    5.2     TEST 2          DATA RELIABILITY.
1193
1194                                                              1. THE TAPE IS INITIATED WITH THE FOLLOWING COMMANDS:
1195                                                                    SET CHARACTERISTIC 40
1196                                                                    REWIND
1197                                                                    WRITE/VERIFY 31 RECORDS OF RANDOM LENGTH AND DAT
1198                                                              2. WRITE AND READ COMMANDS ARE SELECTED AT RANDOM AND
1199                                                                 EXECUTED A RANDOM NUMBER OF TIMES WITH RANDOM
1200                                                                 LENGTHS AND RANDOM PATTERN UNTIL END OF TAPE IS REA
1201                                                              3. AT THE END OF EACH PASS, A REWIND COMMAND IS ISSUED
1202                                                                 A PERFORMANCE REPORT IS PRINTED.
1203
1204                                                                    NOTE:   IF A RESTART COMMAND IS USED TO INITIATE
1205                                                                            TEST 1, THE INITIAL REWIND COMMAND IS NO
1206
1207
1208                                    5.3     TEST 3          WRITE COMPATABILITY/WRITE UTILITY.
1209
1210                                                            REWINDS AND WRITES RECORDS OF RANDOM LENGTHS
1211                                                            AND RANDOM DATA FROM BOT TO EOT.
1212
1213
1214                                    5.4     TEST 4          READ COMPATABILITY/READ UTILITY.
1215
1216                                                            REWINDS AND READS ENTIRE TAPE, FORWARD AND REVERSE.
1217
1218
1219                                    5.5     TEST 5          EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.
1220
1221                                                            THE SEQUENCE OF COMMANDS ENTERED BY THE OPERATOR
1222                                                            IS EXECUTED.  IF NO COMMANDS WERE ENTERED, A
1223                                                            DEFAULT SEQUENCE OF REWIND/WRITE/READ REV/READ FWD/REWIN
1224                                                            OF ENTIRE TAPE IS EXECUTED WITH RANDOM PATTERN
1225                                                            AND RECORD LENGTH OF 2048 BYTES.
```

6.0 DEVICE INFORMATION TABLES

6.1 GENERAL

THE TS04 TAPE SUBSYSTEM CONSISTS OF A TS11 UNIBUS TO SERIAL BUS CONTROLLER CONNECTED TO A TS04 DRIVE. FROM A SOFTWARE VIEWPOINT THIS CONFIGURATION IS UNIQUE (FOR A UNIBUS DEVICE) IN A NUMBER OF WAYS:

A. ONLY ONE REGISTER MAY BE WRITTEN - TSDB (TAPE SYSTEM DATA BUFFER).

B. TWO REGISTERS MAY BE READ TSSR AND TSBA (TAPE SYSTEM STATUS REGISTER AND TAPE SYSTEM BUS ADDRESS REGISTER).

C. COMMANDS ARE NOT WRITTEN TO THE DRIVE; RATHER, COMMAND POINTERS ARE WRITTEN WHICH POINT TO COMMAND PACKETS SOME WHERE IN CPU MEMORY. THE COMMAND POINTER IS USED BY THE TS04 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

D. THE TSSR CONTAINS ALL THE INFORMATION WHICH WILL BE NEC ESSARY 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.

E. EXTENDED STATUS REGISTERS ARE NOT READ DIRECTLY FROM DRIVE REGISTERS; RATHER, A "GET STATUS" COMMAND IS ISSUED WHICH WILL CAUSE THE TS04 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. SEE 6.3.

F. 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.

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

6.2 UNIBUS INTERFACE SPECIFICATIONS

TS11/ TS04	INT. VECTOR	UNIBUS ADDRESS	REGISTER
FIRST	224	772520	TSBA/TSDB
		772522	TSSR
SECOND	154	772524	TSBA/TSDB
		772526	TSSR
THIRD	160	772530	TSBA/TSDB
		772532	TSSR
FOURTH	164	772534	TSBA/TSDB
		772536	TSSR

6.3 BIT DEFINITIONS FOR TS11/TS04 REGISTERS

6.3.1 TS11/TS04 REGISTER SUMMARY

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	
TSBA	A15	A14	A13	A12	A11	A10	A09	A08	A07	A06	A05	A04	A03	A02	A0	
TSDB	P15	P14	P13	P12	P11	P10	P09	P08	P07	P06	P05	P04	P03	P02	P1	
TSSR	SC	UPE	SPE	RMR	NXM	NBA	A17	A16	SSR	OFL	FC1	FC0	TC2	TC1	TC	
XST0	TMK	RLS	LET	RLL	WLE	NEF	ILC	ILA	MOT	ONL	IE	VCK	PED	WLK	BO	
XST1	DLT		COR	CRS	TIG	DBF NZO	SCK		IPR	SYN DRP	IPO ITM	IED LCO	POS NZN	POL LRC	UN CR	
XST2	OPM	SIP	BPE	CAF		WCF		DTP	DT7	DT6	DT5	DT4	DT3	DT2	DT	
XST3	MICRO DIAGNOSTIC ERROR CODE								LMX	OPI	REV	CRF	DCK	NOI	LX	

TERMINATION CLASS CODES (TSSR TC0-TC2):

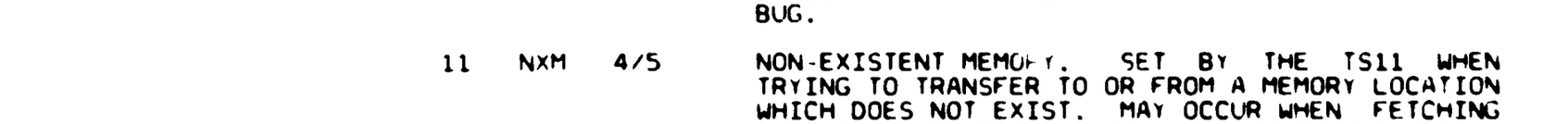
0 = NORMAL TERMINATION
1 = ATTENTION CONDITION
2 = TAPE STATUS ALERT
3 = FUNCTION REJECT
4 = RECOVERABLE ERROR - TAPE POSITION = ONE RECORD DOWN TAPE FROM START OF FUNCTION
5 = RECOVERABLE ERROR - TAPE NOT MOVED
6 = UNRECOVERABLE ERROR - TAPE POSITION LOST
7 = FATAL CONTROLLER ERROR

FATAL CLASS CODES (TSSR FC0-FC1):

0 = MICRO DIAGNOSTIC FAILURE (DISPLAYED IN TS04 OPERATOR PANEL AND
1 = I/O SEQUENCER CROM PARITY ERROR.
2 = MICROPROCESSOR CROM PARITY ERROR.
SILO PARITY ERROR.
SERIAL BUS PARITY ERROR DETECTED AT TS11 (SPE).
SERIAL BUS PARITY ERROR DETECTED AT TS04 (BPE).
FATAL ERROR HALTS 1750 1777 IN TS04 PROGRAM COUNTER DISPLAY.
3 LOSS OF AC POWER HAS BEEN DETECTED.

6.3.2 TS11 STATUS REGISTER (TSSR)

UNIBUS ADDRESS + 2 - READ ONLY



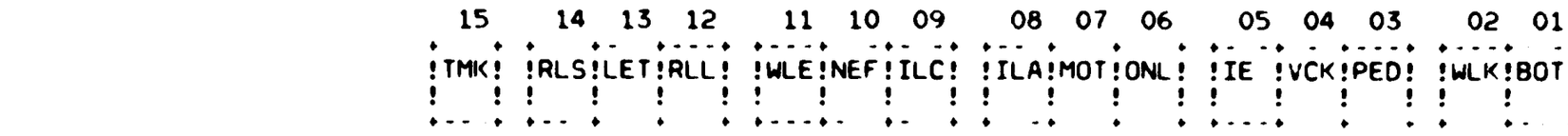
BIT	NAME	TCC	DEFINITION
15	SC	S	SPECIAL CONDITION. WHEN SET, INDICATES THAT THE LAST COMMAND DID NOT COMPLETE WITHOUT INCIDENT. SPECIFICALLY, EITHER AN ERROR WAS DETECTED OR AN EXCEPTION CONDITION OCCURRED. EXCEPTION CONDITIONS CAN BE TAPE MARKS ON READ COMMANDS, REVERSE MOTION AND AT BOT, EOT WHILE WRITING, ETC. MAY ALSO BE SET BY THE ERROR BITS CONTAINED IN THE TSSR REGISTER: UPE, SPE, RMR, AND NXM. THE TERMINATION CLASS BITS ARE SOME OTHER THAN 0 (UNLESS RMR IS THE ONLY ERROR - SEE RM
14	UPE	4/5	UNIBUS PARITY ERROR. SET BY THE TS11 WHEN IT DETECTS A PARITY ERROR ON THE UNIBUS DATA WHEN TRANSFERRING TO OR FROM THE CPU'S MEMORY.
13	SPE	7	SERIAL BUS PARITY ERROR. THIS BIT IS SET BY THE TS11 WHEN IT DETECTS A SERIAL BUS PARITY ERROR ON DATA RECEIVED FROM THE TS04.
12	RMR	S	REGISTER MODIFICATION REFUSED. SET BY THE TS11 WHEN A COMMAND POINTER IS LOADED INTO TSDB AND SUB-SYSTEM READY (SSR) IS NOT SET. NOTE THAT THIS BIT CAUSES SPECIAL CONDITION BUT NO TERMINATIO CLASS (IN FACT, THE TS04 NEVER SEES THIS ERROR) BECAUSE ON A SYSTEM WITH NO BUGS, THIS BIT MAY COME UP ON AN ATTENTION MESSAGE. IF ATTNS ARE NOT ENABLED, THIS BIT COMING UP IS AN INDICATION OF EITHER A FATAL CONTROLLER ERROR OR A SOFTWARE BUG.
11	NXM	4/5	NON-EXISTENT MEMORY. SET BY THE TS11 WHEN TRYING TO TRANSFER TO OR FROM A MEMORY LOCATION WHICH DOES NOT EXIST. MAY OCCUR WHEN FETCHING THE COMMAND PACKET, FETCHING OR STORING DATA, OR STORING THE MESSAGE PACKET.
10	NBA	S	NEED BUFFER ADDRESS. WHEN SET, INDICATES THAT THE TS04 NEEDS A MESSAGE BUFFER ADDRESS. THIS

				BIT IS CLEARED DURING THE SET CHARACTERISTICS COMMAND (IF A GOOD ADDRESS WAS GIVEN).
	09	A17	S	BUS ADDRESS BIT 17. A17 AND A16 (BIT 08) TRACK THE VALUES OF BITS 17 AND 16 OF THE TSBA REGISTER.
	08	A16	S	BUS ADDRESS BIT 16. SEE A17 (BIT 09).
	07	SSR	S	SUB SYSTEM READY. WHEN SET, INDICATES THAT THE TS11/TS04 SUBSYSTEM IS NOT BUSY AND IS READY TO ACCEPT A NEW COMMAND POINTER.
	06	OFL	S,1,3	OFF-LINE. WHEN SET, INDICATES THAT THE TS04 IS OFF LINE AND UNAVAILABLE FOR ANY TAPE MOTION COMMANDS. THIS BIT CAN CAUSE A TERMINATION CLASS OF 1 (ON ATTN INTERRUPT) OR 3 (RESULTS IN NEF).
	05	FC1	7	FATAL TERMINATION CLASS 01. FC1 AND FC0 (BIT 04) ARE USED TO INDICATE THE TYPE OF FATAL ERROR WHICH HAS OCCURRED ON THE TS04. THESE BITS ARE VALID ONLY WHEN SC IS SET AND THE TERMINATION CLASS CODE BITS ARE ALL SET (111).
	04	FC0	7	FATAL TERMINATION CLASS 00. SEE FC1 (BIT 05).
	03	TC2	S	TERMINATION CLASS BIT 02. THIS BIT, ALONG WITH THE TC1 AND TC0 BITS, ACT AS AN OFFSET VALUE WHENEVER AN ERROR OR EXCEPTION CONDITION OCCURS ON A COMMAND. EACH OF THE EIGHT POSSIBLE VALUES OF THIS FIELD REPRESENT A PARTICULAR CLASS OF ERRORS OR EXCEPTIONS. THE CONDITIONS IN EACH CLASS HAVE SIMILAR SIGNIFICANCE AND, AS APPLICABLE, RECOVERY PROCEDURES. THE CODE PROVIDED IN THIS FIELD IS EXPECTED TO BE UTILIZED AS AN OFFSET INTO A DISPATCH TABLE FOR HANDLING OF THE CONDITION.
	02	TC1	S	TERMINATION CLASS BIT 01. SEE TC2 (BIT 03).
	01	TC0	S	TERMINATION CLASS BIT 00. SEE TC2 (BIT 03).
	00			NOT USED.

UNIBUS ADDRESS + 2 WRITE ONLY

SUBSYSTEM INITIALIZE

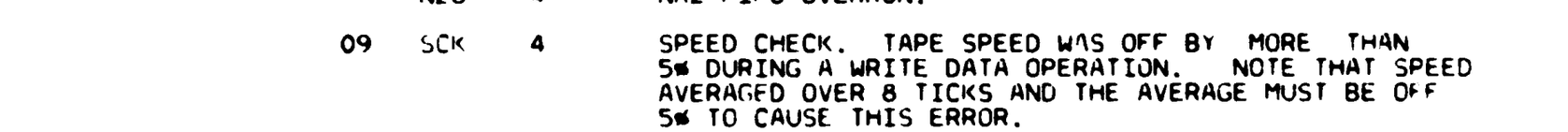
6.3.3 EXTENDED STATUS REGISTER 0 (XSTAT0)



BIT	NAME	TCC	DEFINITION
15	TMK	S,2	TAPE MARK DETECTED. SET WHENEVER A TAPE MARK WAS DETECTED DURING A READ, SPACE, OR SKIP COMMAND AND AS A RESULT OF THE WRITE TAPE MARK OR WITE TAPE MARK RETRY COMMANDS.
14	RLS	2	RECORD LENGTH SHORT. THIS BIT INDICATES THAT EITHER THE RECORD'S LENGTH WAS SHORTER THAN THE BYTE COUNT ON READ OPERATIONS, A SPACE RECORD OPERATION ENCOUNTERED A TAPE MARK OR BOT BEFORE THE POSITION COUNT WAS EXHAUSTED, OR A SKIP TAPE MARKS COMMAND WAS TERMINATED BY ENCOUNTERING BOT OR A DOUBLE TAPE MARK (IF THAT OPERATIONAL MODE IS ENABLED, SEE LET) PRIOR TO EXHAUSTING THE POSITION COUNTER.
13	LET	2	LOGICAL END OF TAPE. SET ONLY ON THE SKIP TAPE MARKS COMMAND WHEN EITHER TWO CONTIGUOUS TAPE MARKS ARE DETECTED OR WHEN MOVING OFF OF BOT AND THE FIRST RECORD ENCOUNTERED IS A TAPE MARK. THE SETTING OF THIS BIT WILL NOT OCCUR UNLESS THIS MODE OF TERMINATION IS ENABLED THROUGH USE OF THE SET CHARACTERISTICS COMMAND.
12	RLL	2	RECORD LENGTH ONG. WHEN SET, THIS BIT INDICATES THAT THE RECORD READ WAS LONGER THAN THE BYTE COUNT SPECIFIED.
11	WLE	3,6	WRITE LOCK ERROR. WHEN SET, INDICATES THAT A WRITE OPERATION WAS ISSUED BUT THE MOUNTED TAPE DID NOT CONTAIN A WRITE ENABLE RING OR THE WRT LOCK SWITCH ACTIVATED DURING THE OPERATION.
10	NEF	3	NON-EXECUTABLE FUNCTION. WHEN SET, INDICATES THAT THE COMMAND COULD NOT BE EXECUTED DUE TO ONE OF THE FOLLOWING CONDITIONS: THE COMMAND SPECIFIED REVERSE TAPE DIRECTION BUT THE TAPE WAS ALREADY POSITIONED AT BOT. THE ISSUING OF ANY COMMAND, EXCEPT REWIND,

1517				
1518				
1519				UNLOAD, OR A COMMAND WITH THE CLEAR VOLUME
1520				CHECK (CVC) BIT SET, WHEN THE VOLUME CHECK
1521				BIT IS SET.
1522				- ANY COMMAND, EXCEPT GET STATUS OR DRIVE
1523				INITIALIZE, WHEN THE TS04 IS OFF-LINE.
1524				- ANY WRITE COMMAND WHEN THE TAPE DOES NOT
1525				CONTAIN A WRITE ENABLE RING (WRITE LOCK
1526				STATUS - WLS).
1527				
1528	09	ILC	3	ILLEGAL COMMAND. SET WHEN A COMMAND IS ISSUED
1529				AND EITHER ITS COMMAND FIELD OR ITS COMMAND
1530				MODE FIELD CONTAINS CODES WHICH ARE NOT
1531				SUPPORTED BY THE TS04.
1532				
1533	08	ILA	3	ILLEGAL ADDRESS. (MORE THAN 18 BITS OR ODD WHEN
1534				AN EVEN ADDRESS IS REQUIRED.)
1535				
1536	07	MOT	S	TAPE IS MOVING.
1537				
1538	06	ONL	S	ON LINE. WHEN SET, INDICATES THAT THE TS04 IS
1539				ON-LINE AND OPERABLE.
1540				
1541	05	IE	S	INTERRUPT ENABLE. REFLECTS THE STATE OF THE
1542				INTERRUPT ENABLE BIT SUPPLIED ON THE LAST
1543				COMMAND.
1544				
1545	04	VCK	S	VOLUME CHECK. WHEN SET, INDICATES THAT THE
1546				DRIVE HAS BEEN EITHER POWERED DOWN OR TURNED
1547				OFF-LINE. CLEARED BY THE CLEAR VOLUME CHECK
1548				(CVC) BIT IN THE COMMAND HEADER WORD. THIS BIT
1549				CAN CAUSE A TERMINATION CLASS OF 3.
1550				
1551	03	PED	S	PHASE ENCODED DRIVE. WHEN SET, INDICATES THAT
1552				THE TS04 IS CAPABLE OF READING AND WRITING ONLY
1553				1600 BPI PHASE ENCODED DATA. WHEN RESET,
1554				INDICATES THAT THE TS04 HAS ONLY 800 BPI NRZI
1555				DATA CAPABILITIES.
1556				
1557	02	WLK	S,3	WRITE LOCKED. WHEN SET, INDICATES THAT THE
1558				MOUNTED REEL OF TAPE DOES NOT HAVE A
1559				WRITE-ENABLE RING INSTALLED. THE TAPE IS,
1560				THEREFORE, WRITE PROTECTED.
1561				
1562	01	BOT	S,3	BEGINNING OF TAPE. WHEN SET, INDICATES THAT
1563				THE TAPE IS POSITIONED AT THE LOAD POINT AS
1564				DENOTED BY THE BOT REFLECTIVE STRIP ON THE
1565				TAPE.
1566				
1567	00	EOT	S,2	END OF TAPE. THIS BIT IS SET WHENEVER THE TAPE
1568				IS POSITIONED AT OR BEYOND THE END OF TAPE
1569				REFLECTIVE STRIP. DOES NOT RESET UNTIL THE
1570				TAPE PASSES OVER THE REFLECTIVE STRIP IN THE
1571				REVERSE DIRECTION UNDER PROGRAM CONTROL.

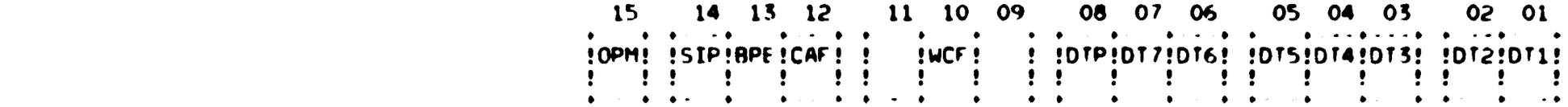
6.3.4 EXTENDED STATUS REGISTER 1 (XSTAT1)



BIT	NAME	TCC	DEFINITION
15	DLT	4	DATA LATE. SET WHEN THE I/O SILO IS FULL ON A READ OR EMPTY ON A WRITE. THESE CONDITIONS OCCUR WHENEVER THE UNIBUS LATENCY EXCEEDS THE DATA TRANSFER RATE OF THE TS04.
14	-	-	NOT USED.
13	COR	S	CORRECTABLE DATA. IN PHASE ENCODED MODE, A CORRECTABLE DATA ERROR HAS BEEN ENCOUNTERED.
12	CRS	4	CREASE DETECTED. FOR NRZI, ALL DATA TRACKS DROPPED OUT FOR MORE THAN THREE CHARACTER TIMES BUT FOR LESS THAN .1 INCHES OF TAPE. FOR PE, EIGHT OUT OF NINE DATA TRACKS WENT DEAD FOR LESS THAN .1 INCHES BEFORE A VALID POSTAMBLE WAS DETECTED.
11	TIG	4	TRASH IN THE GAP. NON-ERASED DATA WAS DETECTED IN A GAP DURING A READ, WRITE, WRITE TAPE MARK, OR ERASE COMMAND.
10	DBF	4	DESKEW BUFFER FAIL. ONE OF THE DESKEW BUFFERS FAILED TO ASSERT "OUTPUT READY" WITHIN 20 MICROSECONDS AFTER BEING ENABLED. THE DEAD TRACK BITS WILL INDICATE ON WHICH TRACKS THIS FAILURE OCCURRED. THIS ERROR IS PROBABLY A RESULT OF A BROKEN FOR MATTER.
	NZO	4	NRZ FIFO OVERRUN.
09	SCK	4	SPEED CHECK. TAPE SPEED WAS OFF BY MORE THAN 5% DURING A WRITE DATA OPERATION. NOTE THAT SPEED AVERAGED OVER 8 TICKS AND THE AVERAGE MUST BE OFF 5% TO CAUSE THIS ERROR.
08			NOT USED.
07	IPR	S,4	INVALID PREAMBLE. SET ON A PE DRIVE IF THE PREAMBLE APPEARS TO BE SHORTER THAN 36 CHARACTERS OR LONGER THAN 44 CHARACTERS. ALSO

```
1628                                                    SET IF THE PREAMBLE IS INCORRECTLY ENCODED
1629                                                    BEYOND THE FIFTEENTH CHARACTER IN READ OR THE
1630                                                    TENTH CHARCTER IN READ AFTER WRITE.
1631
1632                                06  SYN   4        SYNCH FAILURE.  SET ON A PE DRIVE IF THE
1633                                                    FORMATTER WAS UNABLE TO ACHIEVE SYNCHRONIZATION
1634                                                    IN THE PREAMBLE.
1635
1636                                    DRP   4        NRZ RECORD DROPPED A CHARACTER (THE NEXT CHARACTER
1637                                                    WAS TO BE CONSIDERED CRC).
1638
1639                                05  IPO   S,4      INVALID POSTAMBLE.  SET ON A PE DRIVE DURING
1640                                                    READ OR WRITE IF ANY OF THE FIRST 39 CHARACTERS
1641                                                    OF THE POSTAMBLE ARE NOT READ CORRECTLY.
1642
1643                                    ITM   S,4      ILLEGAL TAPE MARK FOR NRZ.
1644
1645                                04  IED   4        INVALID END DATA.  FOR PE, EIGHT OUT OF NINE
1646                                                    TRACKS WENT DEAD BEFORE THE POSTAMBLE WAS DETECTED.
1647
1648                                    LRO   4        FOR NRZI, DATA WAS NOT DETECTED IN EITHER THE
1649                                                    LRCC OR CRCC WINDOWS. (LRC WAS ZERO)
1650
1651                                03  POS   S,4      POSTAMBLE SHORT.  SET ON PE DRIVES DURING A
1652                                                    READ OR WRITE WHEN LESS THAN 38 ALL ZEROES
1653                                                    CHARACTERS ARE READ FOLLOWING THE ALL ONES
1654                                                    CHARACTER.
1655
1656                                    NZN   S,4      NRZ NOISE RECORD (FEWER THAN 13(10) FRAMES).
1657
1658                                02  POL   4        POSTAMBLE LONG.  SET ON PE DRIVES DURING READ
1659                                                    OR WRITE OPERATIONS WHEN THE POSTAMBLE EXCEEDS
1660                                                    42 CHARACTERS.
1661
1662                                    LRC   4        LRC ERROR.  SET ON NRZI DRIVES WHEN THE LRCC
1663                                                    CHARACTER WAS FOUND IN ERROR.
1664
1665                                01  UNC   4        UNCORRECTABLE DATA.  SET ON PE DRIVES WHEN A
1666                                                    PARITY ERROR OCCURRED WITHOUT A CORRESPONDING
1667                                                    DEAD TRACK INDICATION.
1668
1669                                    CRC   4        CRC ERROR.  SET ON NRZI DRIVES WHEN THE CRC
1670                                                    CHARACTER WAS FOUND TO BE IN ERROR.
1671
1672                                00  MTE   4        MULTI-TRACK ERROR.  SET ON PE DRIVES WHEN MORE
1673                                                    THAN ONE DEAD TRACK OCCURRED IN THE PREAMBLE OR
1674                                                    IN THE DATA FIELD.
1675
1676                                    VPE   4        VERTICAL PARITY ERROR.  SET ON NRZI DRIVES WHEN
1677                                                    A CHARACTER DID NOT CONTAIN AN ODD NUMBER OF
1678                                                    ONE BITS.
```

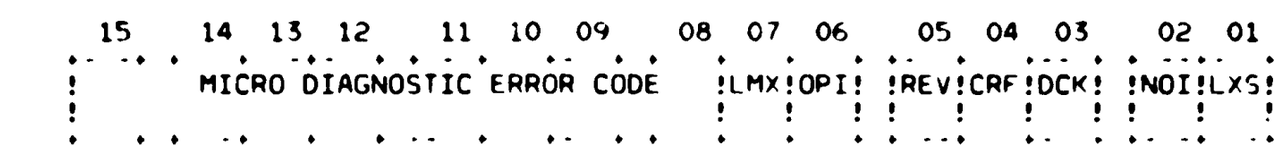
6.3.5 EXTENDED STATUS REGISTER 2 (XSTAT2)



BIT	NAME	TCC	DEFINITION
15	OPM	S	OPERATION IN PROGRESS. (TAPE MOVING)
14	SIP	7,F2	SILO PARITY ERROR. CAUSES FATAL CLASS 2 BECAUSE THE ERROR MIGHT HAVE OCCURRED DURING THE TRANSMISSION OF THE MESSAGE PACKET.
13	BPE	7,F2	SERIAL BUS PARITY ERROR AT DRIVE. SET BY THE TS04 WHEN A PARITY ERROR IS DETECTED ON DATA TRANSMITTED FROM THE TS11 TO THE TS04. CAUSES FATAL CLASS 2 BECAUSE THE ERROR MIGHT HAVE OCCURRED DURING THE TRANSMISSION OF THE MESSAGE PACKET.
12	CAF	7	CAPSTAN ACCELERATION FAIL. AFTER ACCELERATING TAPE FOR .2 INCHES, THE TAPE SPEED WAS CHECKED AND FOUND TO BE OUT OF TOLERANCE BY MORE THAN 10%.
11			NOT USED.
10	WCF	7	THE WRITE BOARD IS NOT EMPTYING THE I/O SILO AT THE PROPER RATE. THIS ERROR CAN BE THE RESULT OF THE WRITE BOARD CLOCK NOT BEING TURNED ON (BROKEN HARDWARE).
09			NOT USED.
08	DTP	S	DEAD TRACK PARITY. THE BITS DTP THROUGH DT0 INDICATE WHICH TRACK(S) WENT DEAD, IF ANY, DURING THE LAST DATA TRANSFER OPERATION. IF DESKEW BUFFER FAIL (DBF) IS SET, THESE BITS INDICATE WHICH CHANNEL FAILED.
07	DT7	S	DEAD TRACK 7. SEE DTP.
06	DT6	S	DEAD TRACK 6. SEE DTP.
05	DT5	S	DEAD TRACK 5. SEE DTP.
04	DT4	S	DEAD TRACK 4. SEE DTP.
03	DT3	S	DEAD TRACK 3. SEE DTP.

```
1735
1736                                    02   DT2    S       DEAD TRACK 2.  SEE DTP.
1737
1738                                    01   DT1    S       DEAD TRACK 1.  SEE DTP.
1739
1740                                    00   DT0    S       DEAD TRACK 0.  SEE DTP.
1741
1742                                NOTE:  ON A SET CHARACTERISTICS COMMAND, THE UCODE LEVEL  IS  RETURNED
1743                                       IN DT7 THRU DT0.  ON A GET STATUS COMMAND, THE RESIDUAL CAPSTAN
1744                                       TICK COUNT (INTERNALLY R7) IS RETURNED IN DT7 THRU DT0.
```

6.3.6 EXTENDED STATUS REGISTER 3 (XSTAT3)



BIT	NAME	TCC	DEFINITION
15 TO 08			MICRO DIAGNOSTIC ERROR CODE. (SEE LIST OF CODES BELOW). ALL ERROR CODES IN THE TABLE WILL BE DISPLAYED ON THE TS04 CONTROL PANEL BUT ONLY CODES HIGHER THAN 110 WILL BE AVAILABLE TO CPU DIAGNOSTICS FOR PRINTOUT IN THE MICRO DIAGNOSTIC ERROR CODE FIELD OF XSTAT3. THIS ERROR CODE FIELD IS VALID ONLY WHEN THE TERMINATION CLASS CODE IN THE TSSR EQUALS 7 AND THE FATAL CLASS CODE IN THE TSSR EQUALS 0, INDICATING AN INTERNAL DIAGNOSTIC FAILURE.
07	NTL	6	LIMIT EXCEEDED. SET WHEN THE TAPE TENSION ARMS HAVE EXCEEDED THEIR ALLOWABLE TRAVEL AND HAVE CAUSED THE ACTIVITATION OF THE LIMIT SWITCHES. NO TENSION EXISTS ON THE MOUNTED TAPE.
06	OPI	6	OPERATION INCOMPLETE. SET WHEN A READ, SPACE, OR SKIP OPERATION HAS MOVED 25 FEET OF TAPE WITHOUT DETECTING ANY DATA ON THE TAPE.
05	REV	5	DIRECTION OF CURRENT OPERATION WAS REVERSE (BUT IS 0 IF REWIND OR FORWARD)
04	CRF	7	CAPSTAN RESPONSE FAILURE. A MOTION COMMAND WAS GIVEN TO THE CAPSTAN BUT WE DID NOT GET A TICK BACK WITHIN A REASONABLE AMOUNT OF TIME.
03	DCK	5,6	DENSITY CHECK. SET ON PE DRIVES WHEN A PE IDENTIFICATION BURST WAS NOT DETECTED WHEN MOVING OFF OF BOT. SET ON NRZI DRIVES WHEN A NON NRZI IDENTIFICATION BURST WAS FOUND WHEN MOVING OFF OF BOT.
02	NOI	6	NOISE RECORD. SET DURING A READ OR SPACE OPERATION WHEN A BURST OF FLUX CHANGES, WHICH DO NOT QUALIFY AS A RECORD (BUT TOO MANY TO IGNORE), ARE DETECTED: NRZI: AT LEAST TWO CHARACTERS IN A ROW BUT LESS THAN TWELVE, FOLLOWED BY A CHARACTER IN EITHER THE CRCC OR LRCC WINDOWS.

```
                                                PE:  AT LEAST 24 CHARACTERS IN A ROW THAT  DO
                                                NOT   QUALIFY  AS  A  TAPE  MARK  OR  A  DATA
                                                PREAMBLE.

                                    01   LXS    S       LIMIT EXCEEDED STATICALLY.  THIS BIT IS SET ANY
                                                        TIME THE LIMIT SWITCHES ARE EXCEEDED.  THIS BIT
                                                        CAN ONLY BE CLEARED BY MANUALLY LOADING THE TAPE.



                                    00   RIB    2       REVERSE INTO  BOT.   A  READ,  SPACE,  OR  SKIP
                                                        COMMAND ALREADY IN PROGRESS HAS ENCOUNTERED THE
                                                        BOT MARKER WHEN  MOVING  TAPE  IN  THE  REVERSE
                                                        DIRECTION.    TAPE MOTION WILL BE HALTED AT BOT.



                        MICRO DIAGNOSTIC ERROR CODES
                        - - -  - - ----   -- - - --  -

                        FOLLOWING IS A LIST OF THE ERRORS WHICH ARE DISPLAYED IN THE MICRO
                        DIAGNOSTIC ERROR CODE (XSTAT3 BITS 15 - 08) AND ALSO IN THE LIGHTS ON THE
                        TS04 CONTROL PANEL, DUE TO FAILURES ON THE CAPSTAN BOARD, I/O BOARDS,
                        WRITE BOARD, READ BOARD, OR FORMATTER BOARD. THE MICRO WILL BE IN A
                        TIGHT LOOP IN THE DISPM PROGRAM, WAITING FOR OPERATOR OR CPU
                        INTERVENTION WHILE THE ERROR IS BEING DISPLAYED IN THE CONSOLE
                        LIGHTS. IT IS APPARENT THAT AN ERROR IS BEING DISPLAYED IF THE "UOK"
                        LIGHT IS NOT LIGHTED, THE PROCESSOR IS NOT STOPPED, AND AN OCTAL
                        NUMBER (100-377) IS BEING DISPLAYED IN THE LIGHTS. TO SCOPE LOOP
                        THESE  TESTS, ENTER MAINTENANCE MODE (ON-LINE SWITCH TO "OFF"
                        POSITION, MAINTENANCE SWITCH UP, PRESS RESET), ENTER THE OFF-LINE
                        TEST NUMBER (SEE SCOPE LOOP COLUMN BELOW) IN THE OPERATOR CONSOLE
                        LIGHTS (ENTER ONE, WITH LEFT MOST SWITCH, ENTER  ZEROES  WITH
                        RIGHT-MOST SWITCH), AND PRESS ON LINE BUTTON. TEST WILL LOOP UNTIL
                        ON-LINE SWITCH IS RETURNED TO OFF LINE POSITION, ERRORS WILL BE
                        DISPLAYED CONTINUOUSLY.

                        ERROR   PROGRAM                 ERROR DESCRIPTION                   LIKELY   SCOPE
                        (DISPLAY)                                                           MODULE   LOOP

                        337     OPERATIONAL             CAPSTAN RUNAWAY ERROR (M3.RNY). CAPSTAN DIDN'T
                                   CODE                 STOP WITHIN ACCEPTABLE WINDOW AFTER LAST COMMAND.



                        100     IOTSM                   BASIC I/O MICRO FAILURE (PARITY         M8966    14
                                                        ERROR, IOATN, HANDSHAKING, AND
                                                        DATA WINDOW TEST BETWEEN THE I/O
                                                        AND MAIN MICROS.

                                                        NOTE:  CAN ALSO BE CAUSED BY THE
                                                               SERIAL BUS .SMIN (SHIFT IN)
                                                               STUCK ASSERTED.

                        101     IOTSM                   ERROR IN I/O CONTROL REGISTER TEST      M8966    15
                                                                                                M8967
```

102	IOTSM	FAILURE OF FRAME COUNTER TEST	M8966	15
103	IOTSM	FAILURE OF I/O SILO NON PARITY ERROR DATA TEST OR THE WRITE FLAG.	M8966 M8963	16
104	IOTSM	FAILURE OF I/O SILO PARITY ERROR TEST OR DATA LATE TEST.	M8966	17
105	IOTSM	FAILURE OF SHIFT LOOP WITH ZEROES.	M8965	20
106	IOTSM	FAILURE OF SHIFT LOOP WITH ONES.	M8965	21
107	IOTSM	FAILURE OF SHIFT LENGTH MUX.	M8965	22
110	IOTSM	FAILURE TO RECEIVE CORRECT OP-CODE FROM TS11 WHEN WE SENT DATA OVER THE SERIAL BUS.	M8965 TS11 MOTHER BD SBUS CABLE	47
111	CATSM	FAILURE OF 1 KHZ CLOCK TEST. TSTS TAC SYNC FLOP AND ATTN, TOO.	G159 CBUS CABLE M8963	2
112	CATSM	LIGHT REGISTER CHANGED WHEN MOTION REGISTER WAS CLEARED.	G159	3,4
113	CATSM	FWD OR MVG BITS WRONG AFTER 1 TICK OF SIMULATED COMMAND AND TACH PULSES.	G159	3,4
114	CATSM	FAILURE OF SIMULATED CAPSTAN SPEED TEST. THE CAPSTAN SPEED COUNTER WAS OUT OF RANGE WHEN TAPE MOTION AT SPEED WAS SIMULATED.	G159	3,4
115	CATSM	FAILURE OF SIMULATED SLOW CAPSTAN TEST. SPEED COUNTER DID NOT LATCH UP WITH MAX COUNT WHEN SLOW TACH TICKS WERE SIMULATED.	G159	3,4
116	CATSM	FAILURE OF SIMULATED CAPSTAN DECEL TEST. COUNTER NOT ZERO FOR FORWARD OR 377 FOR REVERSE WHILE DECELERATING, OR MVG BIT NOT 1.	G159	3,4
117	CATSM	FAILURE OF MOVING FLOP TO GO TO ZERO AFTER STOPPING (DIRECTION REVERSAL FOR ONE TACH TICK).	G159	3,4
120	PETSM	FAILURE OF WRITE BOARD TO TURN ON AND EMPTY THE SILO, OR DATA LATE BIT DOESN'T WORK.	M8929 M8966	23
121	PETSM	FAILURE OF WRITE BOARD TO EMPTY SILO AT CORRECT SPEED.	M8929	23

124	PETSM	FORMATTER FLAG DOESN'T WORK ON THE M8922.	M8922	24	
125	PETSM	FORMATTER SILO FILLING AND DATA ERROR	M8922 M8923 M8924	24	
126	PETSM	PEAK SHIFT TEST ERROR	M8922 M8923 M8924	25	
127	PETSM	FORMATTER TABLE LOOKUP ROM CHECKSUM TEST ERROR	M8922 M8923 M8924	26	

```
                                        .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.
                                        ;

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

                                                HEADER  CZTSH,D.0,5000,1,0INTPRI
002000                                  L$NAME::                ;DIAGNOSTIC NAME
002000      103                                                                 .ASCII /C/
002001      132                                                                 .ASCII /Z/
002002      124                                                                 .ASCII /T/
002003      123                                                                 .ASCII /S/
002004      110                                                                 .ASCII /H/
002005      000                                                                 .BYTE   0
002006      000                                                                 .BYTE   0
002007      000                                                                 .BYTE   0
002010                                  L$REV::                 ;REVISION LEVEL
002010      104                                                                 .ASCII  /D/
002011                                  L$DEPO::                ;0
002011      060                                                                 .ASCII  /0/
002012                                  L$UNIT::                ;NUMBER OF UNITS
002012  000001                                                                  .WORD   T$PTHV
002014                                  L$TIML::                ;LONGEST TEST TIME
002014  005000                                                                  .WORD   5000
002016                                  L$HPCP::                ;POINTER TO H.W. QUES.
002016  025266                                                                  .WORD   L$HARD
002020                                  L$SPCP::                ;POINTER TO S.W. QUES.
002020  025340                                                                  .WORD   L$SOFT
002022                                  L$HPTP::                ;PTR. TO DEF. H.W. PTABLE
002022  002174                                                                  .WORD   L$HW
002024                                  L$SPTP::                ;PTR. TO S.W. PTABLE
002024  002202                                                                  .WORD   L$SW
002026                                  L$LADP::                ;DIAG. END ADDRESS
002026  026746                                                                  .WORD   L$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$DTP::                 ;PTR. TO DISPATCH TABLE
002040  002124                                                                  .WORD   L$DISPA
002042                                  L$PRIO::                ;DIAGNOSTIC RUN PRIORITY
002042  000340                                                                  .WORD   0INTPRI
002044                                  L$ENVI::                ;FLAGS DESCRIBE HOW IT WAS SETUP
```

```
1985  002044  000000                                                  .WORD   0
1986  002046                  L$EXP1::        ;EXPANSION WORD
1987  002046  000000                                                  .WORD   0
1988  002050                  L$MREV::        ;SVC REV AND EDIT #
1989  002050     003                                                  .BYTE   C$REVISI
1990  002051     003                                                  .BYTE   C$EDIT
1991  002052                  L$EF::          ;DIAG. EVENT FLAGS
1992  002052  000000                                                  .WORD   0
1993  002054  000000                                                  .WORD   0
1994  002056                  L$SPC::
1995  002056  000000                                                  .WORD   0
1996  002060                  L$DEVP::        ; POINTER TO DEVICE TYPE LIST
1997  002060  002164                                                  .WORD   L$DVTYP
1998  002062                  L$REPP::        ;PTR. TO REPORT CODE
1999  002062  016150                                                  .WORD   L$RPT
2000  002064                  L$EXP4::
2001  002064  000000                                                  .WORD   0
2002  002066                  L$EXP5::
2003  002066  000000                                                  .WORD   0
2004  002070                  L$AUT::         ;PTR. TO ADD UNIT CODE
2005  002070  021770                                                  .WORD   L$AU
2006  002072                  L$DUT::         ;PTR. TO DROP UNIT CODE
2007  002072  021716                                                  .WORD   L$DU
2008  002074                  L$LUN::         ;LUN FOR EXERCISERS TO FILL
2009  002074  000000                                                  .WORD   0
2010  002076                  L$DESP::        ;POINTER TO DIAG. DESCRIPTION
2011  002076  002136                                                  .WORD   L$DESC
2012  002100                  L$LOAD::        ;GENERATE SPECIAL AUTOLOAD EMT
2013  002100  104035                                                  EMT     E$LOAD
2014  002102                  L$ETP::         ;POINTER TO ERRTBL
2015  002102  000000                                                  .WORD   0
2016  002104                  L$ICP::         ;PTR. TO INIT CODE
2017  002104  017704                                                  .WORD   L$INIT
2018  002106                  L$CCP::         ;PTR. TO CLEAN-UP CODE
2019  002106  021654                                                  .WORD   L$CLEAN
2020  002110                  L$ACP::         ;PTR. TO AUTO CODE
2021  002110  021232                                                  .WORD   L$AUTO
2022  002112                  L$PRT::         ;PTR. TO PROTECT TABLE
2023  002112  017676                                                  .WORD   L$PROT
2024  002114                  L$TEST::        ;TEST NUMBER
2025  002114  000000                                                  .WORD   0
2026  002116                  L$DLY::         ;DELAY COUNT
2027  002116  000000                                                  .WORD   0
2028  002120                  L$HIME::        ;PTR. TO HIGH MEM
2029  002120  000000                                                  .WORD   0
2030
```

```
                                        .SBTTL  DISPATCH TABLE

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

002122                                          DISPATCH 5
002122  000005                                                          .WORD   5
002124                                  L$DISPATCH::
002124  022064                                                          .WORD   T1
002126  023452                                                          .WORD   T2
002130  024126                                                          .WORD   T3
002132  024272                                                          .WORD   T4
002134  024424                                                          .WORD   T5

                                        .SBTTL  DESCRIPTIVE TEXT

                                        ;++
                                        ;2 LINES OF TEXT PRINTED TO THE OPERATOR TO IDENTIFY THE DIAGNOSTIC AND THE DEVI
                                        ;--

002136                                          DESCRIPT        <DATA RELIABILITY TEST>
002136                                  L$DESC::
002136  040504  040524  051040                                          .ASCIZ  /DATA RE
002144  046105  040511  044502
002152  044514  054524  052040
002160  051505  000124
                                                                        .EVEN
002164                                          DEVTYP          <TS11>
002164                                  L$DVTYP::
002164  051524  030461  000                                           .ASCIZ  /TS11/
        002172                                                          .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.
                                        ;--

002172                                          BGNHW   DFPTBL
002172  000002                                                                 .WORD   L10000-L
002174                                  L$HW::
002174                                  DFPTBL::


002174  172522                         172522         ;TSSR ADDRESS.
002176  000224                         224            ;VECTOR ADDRESS.

002200                                          ENDHW
002200                                  L10000:
```

```
                                        .SBTTL  SOFTWARE P-TABLE

                                        ;++
                                        ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
                                        ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
                                        ;--

002200                                          BGNSW   SFPTBL
002200  000043                                                                  .WORD   L10001 ;
002202                                  L$SW::
002202                                  SFPTBL::

002202      001                         CLRFLG:: .BYTE  1               ;CLEAR COUNTERS FLAG.
002203      000                         RRANV::  .BYTE  0               ;RESET RANDOM VARIABLES EACH PASS FLAG.
002204      000                         HAE::    .BYTE  0               ;HALT AFTER EACH COMMAND FLAG.
002205      000                         ERCVER:: .BYTE  0               ;ENABLE RECOVERABLE ERROR PRINTS FLAG.
002206      001                         BADTSW:: .BYTE  1               ;BAD TAPE SWITCH TO REWRITE ON SAME SPOT & DETEC
002207      000                                  .BYTE  0               ;SPARE
002210      000                         DINT::   .BYTE  0               ;DISABLE INTERRUPTS FLAG.
002211      000                         IREC::   .BYTE  0               ;INHIBIT ERROR RECOVERY FLAG.
002212      000                         CHGFLG:: .BYTE  0               ;CHANGE CMD SEQ TABLE FLAG.
002213      000                                  .BYTE  0               ;SPARE.
002214      000                         PIRE::   .BYTE  0               ;INHIBIT RESIDUAL FRAMECOUNT ERROR REPORT FLAG.
002215      000                                  .BYTE  0               ;SPARE.
002216  000040                          CHAR::   CH.EAI                 ;CHARACTERISTICS CODE (DEFAULT = 40).
```

```
2109  002220  000015                CMDD::  .WORD   13.          ;COMMAND 2 (DEFAULT = REWIND).
2110  002222  000001                        .WORD   1            ;BYTE COUNT
2111  002224  000001                        .WORD   1            ;NUMBER OF OPERATIONS
2112  002226  000007                        .WORD   RANP         ;PATTERN
2113  002230  000004                        .WORD   4            ;COMMAND 3  (DEFAULT = WRITE)
2114  002232  004000                        .WORD   DATCNT       ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
2115  002234  076400                        .WORD   32000.       ;NUMBER OF OPERATIONS (DEFAULT = 32000).
2116  002236  000007                        .WORD   RANP         ;PATTERN (DEFAULT = RANDOM).
2117  002240  000003                        .WORD   3            ;COMMAND 4 (DEFAULT = READ REV).
2118  002242  004000                        .WORD   DATCNT       ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
2119  002244  076400                        .WORD   32000.       ;NUMBER OF OPERATIONS (DEFAULT = 32,000).
2120  002246  000007                        .WORD   RANP         ;PATTERN (DEFAULT = RANDOM).
2121  002250  000002                        .WORD   2            ;COMMAND 5 (DEFAULT = READ FWD).
2122  002252  004000                        .WORD   DATCNT       ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
2123  002254  076400                        .WORD   32000.       ;NUMBER OF OPERATIONS (DEFAULT = 32,000).
2124  002256  000007                        .WORD   RANP         ;PATTERN (DEFAULT = RANDOM).
2125  002260  000015                        .WORD   13.          ;COMMAND 6 (DEFAULT = REWIND).
2126  002262  000001                        .WORD   1            ;BYTE COUNT
2127  002264  000001                        .WORD   1            ;NUMBER OF OPERATIONS
2128  002266  000007                        .WORD   RANP         ;PATTERN
2129  002270  000033                        .WORD   27.          ;END OF CMD SEQ TABLE CODE (DEF) OR CMD 7
2130  002272  004000                        .WORD   DATCNT       ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
2131  002274  076400                        .WORD   32000.       ;NUMBER OF OPERATIONS (DEFAULT = 32000).
2132  002276  000007                        .WORD   RANP         ;PATTERN (DEFAULT = RANDOM).
2133  002300  000033                        .WORD   27.          ;END OF CMD SEQ TABLE CODE (DEF) OR CMD 8
2134  002302  004000                        .WORD   DATCNT       ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
2135  002304  076400                        .WORD   32000.       ;NUMBER OF OPERATIONS (DEFAULT = 32000).
2136  002306  000007                        .WORD   RANP         ;PATTERN (DEFAULT = RANDOM).
2137
2138  002310                                ENDSW
2139  002310                        .10001:
2140
2141  002310                        ENDMOD
```

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

002310                                          BGNMOD

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

002310                                          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                          PRIO7== 340
```

```
2198            000300                  PRI06== 300
2199            000240                  PRI05== 240
2200            000200                  PRI04== 200
2201            000140                  PRI03== 140
2202            000100                  PRI02=  100
2203            000040                  PRI01== 40
2204            000000                  PRI00== 0
2205                                    ;
2206                                    ;OPERATOR FLAG BITS
2207                                    ;
2208            000004                  EVL=          4
2209            000010                  LOT==        10
2210            000020                  ADR==        20
2211            000040                  IDU==        40
2212            000100                  ISR==       100
2213            000200                  UAM==       200
2214            000400                  BOE==       400
2215            001000                  PNT==      1000
2216            002000                  PRI==      2000
2217            004000                  IXE==      4000
2218            010000                  IBE==     10000
2219            020000                  IER==     20000
2220            040000                  LOE==     40000
2221            100000                  HOE==    100000
2222
2223
2224                                    ; REGISTER USAGE.
2225                                    ;
2226                                    ;       R0 - PASSES PARAMETERS TO/FROM DIAGNOSTIC SUPERVISOR.
2227                                    ;       R1 - COMMAND SEQUENCE TABLE POINTER.
2228                                    ;       R2 - GENERAL PURPOSE REGISTER.
2229                                    ;       R3 - GENERAL PURPOSE REGISTER.
2230                                    ;       R4 - GENERAL PURPOSE REGISTER.
2231                                    ;       R5 - CURRENT LOGICAL DEVICE NUMBER x 2.
2232                                    ;       R6 - STACK POINTER.
2233                                    ;       R7 - PROGRAM COUNTER.
2234
2235                                    ;THE FOLLOWING ARE BIT DEFINITIONS FOR THE TSSR REGISTERS.
2236
2237            100000                  TS.SC==100000           ;SPECIAL CONDITION BIT.
2238            040000                  TS.UPE==40000           ;UNIBUS PARITY ERROR
2239            020000                  TS.SPE==20000           ;SERIAL BUS PARITY ERROR.
2240            010000                  TS.RMR==10000           ;REGISTER MODIFICATION REFUSED.
2241            004000                  TS.NXM==4000            ;NON-EXISTENT MEMORY.
2242            002000                  TS.NBA==2000            ;NEED BUFFER ADDRESS.
2243            001000                  TS.A17==1000            ;BUS ADDRESS BIT 17.
2244            000400                  TS.A16==400             ;BUS ADDRESS BIT 16.
2245            000200                  TS.SSR==200             ;UNIT READY BIT.
2246            000100                  TS.OFL==100             ;OFF LINE.
2247            177717                  TSC.FCC==177717         ;FATAL CLASS CODE MASK.
2248            177761                  TSC.TCC==177761         ;TERMINATION CLASS CODE MASK.
```

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

100000                          ACK.C==100000           ;ACKNOWLEDGE BIT
040000                          CVC.C==40000            ;CLEAR VOLUME CHECK.
020000                          OPP.C==20000            ;OPPOSITE BIT
010000                          SWB.C==10000            ;SWAP BYTE BIT
004000                          MOD.C3==4000            ;MODE BIT 3
004000                          BRF.C==4000             ;BYTE/RECORD/FILE COUNT FLAG BIT.  NOT USED
                                                        ;BY TS04 BUT USED INTERNALLY BY THIS PROGRAM ONL
002000                          MOD.C2==2000            ;MODE BIT 2
001000                          MOD.C1==1000            ;MODE BIT 1
000400                          MOD.C0==400             ;MODE BIT 0
000200                          IE.C==200               ;INTERRUPT ENABLE
000100                          FMT.C1==100             ;FORMAT BIT 1
000100                          VFY.C==100              ;WRITE VERIFY FLAG BIT.  INTERNAL USE ONLY.
                                                        ;NOT USED BY TS04.
000040                          FMT.C0==40              ;FORMAT BIT 0.
000040                          JMP.C==40               ;JUMP BIT TO DIRECT THIS PROGRAM TO JUMP TO
                                                        ;A CERTAIN LOCATION IN THE COMMAND SEQUENCE
                                                        ;TABLE.  INTERNAL USE ONLY.
000020                          CMD.C4==20              ;COMMAND BIT 4
000020                          DLY.C==20               ;INSERT DELAY. INTERNAL USE ONLY.
000010                          CMD.C3==10              ;COMMAND BIT 3
000004                          CMD.C2==4               ;COMMAND BIT 2
000002                          CMD.C1==2               ;COMMAND BIT 1
000001                          CMD.C0==1               ;COMMAND BIT 0

                                ;       BIT DEFINITIONS FOR DEVICE CHARACTERISTICS.

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

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

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

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

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

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

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

```
2305
2306                                    ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 3.
2307
2308    000010                          X3.DCK==10                      ;DENSITY CHECK.
2309    157400                          X3.RNY==157400                  ;CAPSTAN RUNAWAY UDIAG ERROR CODE.
2310
2311                                    ;THE FOLLOWING DEFINITIONS SHOW THE RELATIVE POSITIONS OF THE COMMAND
2312                                    ;PACKET ENTRIES.
2313
2314    000000                          CP.CMD==0                       ;CMDPKT+0==TS04 COMMAND.
2315    000002                          CP.ADL==2                       ;CMDPKT+2==BUFFER ADDRESS LOW.
2316    000004                          CP.ADH==4                       ;CMDPKT+4==BUFFER ADDRESS HIGH.
2317    000006                          CP.CNT==6                       ;CKDPKT+6==BYTE/FILE/RECORD COUNT
2318
2319                                    ;       MISCELLANEOUS DEFINITIONS.
2320
2321    000340                          INTPRI==PRI07                   ;PRIORITY TO BE USED IN INTERRUPT STATE.
2322    002452                          TSBA==TSDB                      ;DATA BUFFER ADDRESS REGISTER.
2323    000010                          SCHCNT==10                      ;ARBITRARY BYTE LENGTH FOR CHARACTERISTIC
2324                                                                    ;BUFFER LENGTH. (EVEN #)
2325    000016                          MSGCNT==16                      ;MESSAGE BUFFER LENGTH IN BYTES. (EVEN #)
2326    003334                          DIABLK==DATAWT                  ;WRITE BUFFER ALSO USED FOR DIAG CMD.
2327    000020                          DIACNT==20                      ;DIAGNOSTIC COMMAND BUFFER EXTENT.
2328    004000                          DATCNT==2048.                   ;MAXIMUM RECORD LENGTH IN BYTES.
2329                                                                    ;THIS COUNT SHOULD BE A MULTIPLE OF 256 TO INSUR
2330                                                                    ;PROPER READ/WRITE BUFFER ALLOCATION BY THE SUPE
2331    000550                          CNTLEN==CNTEND CNTBGN           ;LENGTH OF STATISTICAL COUNTER AREA.
2332    177740                          RNOPSC==177740                  ;RANDOM # OF OPERATIONS MASK.
2333    000007                          RANP==7                         ;CODE TO SELECT RANDOM PATTERN.
2334    000020                          RRECL==16.                      ;READ RECOVERY ATTEMPT LIMIT.
2335    000020                          WRECL==16.                      ;WRITE RECOVERY ATTEMPT LIMIT.
2336    153624                          RANBC==153624                   ;CONSTANT USED TO RESET RANDOM # GENERATOR BASE.
2337    032561                          RANSC==32561                    ;CONSTANT USED TO RESET RANDOM # SAVE LOCATION.
2338    177774                          NINUSE==17 774                  ;NOT IN USE CODE FOR DEVICE STATE TABLE.
2339    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
2340                                                                    ;NOT "COMMAND' BITS.
2341
2342                                    ;THE FOLLOWING DEFINES THE COMMAND WORD FOR EACH TS04 COMMAND.
2343
2344    100013                          DRI==   ACK.C!CMD.C3!CMD.C1!CMD.C0
2345                                                                    ;DRIVE INIT.
2346
2347    104001                          RDF==   ACK.C!BRF.C!CMD.C0
2348                                                                    ;READ FORWARD
2349
2350    104401                          RDR==   ACK.C!BRF.C!MOD.C0!CMD.C0
2351                                                                    ;READ REVERSE
2352
2353    104005                          WRT==   ACK.C!BRF.C!CMD.C0!CMD.C2
2354                                                                    ;WRITE COMMAND
2355
2356    104105                          WTV==   ACK.C!BRF.C!VFY.C!CMD.C0!CMD.C2
2357                                                                    ;WRITE VERIFY
2358
2359    104010                          SRF ==  ACK.C!BRF.C!CMD.C3
2360                                                                    ;SPACE RECORD FORWARD
```

```
2361
2362          104410                          SRR==   ACK.C!BRF.C!MOD.CO!CMD.C3
2363                                                                  ;SPACE RECORD REVERSE
2364
2365          105401                          RNR=    ACK.C!BRF.C!MOD.C1!MOD.CO!CMD.CO
2366                                                          ;READ REV RETRY1 - REREAD NEXT REVERSE, IE. SPACE FWD, READ REVE
2367
2368          125401                          RNF==   ACK.C!BRF.C!OPP.C!MOD.C1!MOD.CO!CMD.CO
2369                                                          ;READ REV RETRY2 - REREAD NEXT FORWARD, IE.READ FORWARD, SPACE R
2370
2371          105001                          RPF=    ACK.C!BRF.C!MOD.C1!CMD.CO
2372                                                          ;READ FWD RETRY1 - REREAD PREVIOUS FORWARD, IE. SPACE REVERSE, R
2373
2374          125001                          RPR==   ACK.C!BRF.C!OPP.C!MOD.C1!CMD.CO
2375                                                          ;READ FWD RETRY2   REREAD PREVIOUS REVERSE, IF. READ REVERSE, SP
2376
2377          105005                          WRR==   ACK.C!MOD.C1!BRF.C!CMD.C2!CMD.CO
2378                                                                  ;WRITE RETRY
2379
2380          102010                          RWD==   ACK.C!MOD.C2!CMD.C3
2381                                                                  ;REWIND COMMAND
2382
2383          100012                          MBR==   ACK.C!CMD.C3!CMD.C1
2384                                                                  ;MESSAGE BUFFER RELEASE
2385
2386          100011                          WTM==   ACK.C!CMD.C3!CMD.CO
2387                                                                  ;WRITE TAPE MARK.
```

```
2388
2389          101011                    WTR==   ACK.C!MOD.C1!CMD.C3!CMD.C0
2390                                                                ;WRITE TAPE MARK RETRY.
2391
2392          105010                    SFF=-   ACK.C!BRF.C!MOD.C1!CMD.C3
2393                                                                ;SPACE FILE FORWARD
2394
2395          105410                    SFR==   ACK.C!BRF.C!MOD.C0!MOD.C1!CMD.C3
2396                                                                ;SPACE FILE REVERSE
2397
2398          100017                    GES=-   ACK.C!CMD.C0!CMD.C1!CMD.C2!CMD.C3
2399                                                                ;GET EXTENDED STATUS
2400
2401          100411                    ERS==   ACK.C!MOD.C0!CMD.C3!CMD.C0
2402                                                                ;ERASE 3 INCHES OF TAPE
2403
2404          100412                    UNL==   ACK.C!MOD.C0!CMD.C3!CMD.C1
2405                                                                ;UNLOAD COMMAND
2406
2407          101012                    CLN==   ACK.C!MOD.C1!CMD.C3!CMD.C1
2408                                                                ;ERASE TAPE.
2409
2410          140004                    SCH==   ACK.C!CVC.C!CMD.C2          ;SET DEVICE CHARACTERISTICS.
2411
2412          100006                    DIA==   ACK.C!CMD.C2!CMD.C1         ;DIAGNOSTICS.
2413
2414          000040                    JMP==   JMP.C                       ;JUMP TO "N'TH COMMAND
2415
2416          000020                    DLY==   DLY.C                       ;DELAY 'N  MS.
2417
2418          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.

                        002310  .       =       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002310  000000          CMDPKT:: 0                              ;1ST WORD IS TS04 COMMAND.
002312  000000                   0                              ;2ND WORD IS THE BUFFER LOW ADDRESS.
002314  000000                   0                              ;3RD WORD IS THE BUFFER HIGH ADDRESS.
002316  000000                   0                              ;4TH WORD IS THE BYTE/RECORD/FILE COUNT.


                                ;       GET STATUS COMMAND PACKET.

                        002320  .       =       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002320  100017          GSCPK::  .WORD   GES


                                ;       MESSAGE BUFFER RELEASE COMMAND PACKET.

                        002324  .       =       .+3&177774      ;MUST BE ON MOD 4 BOUNDRY.
002324  100012          BRCPK::  .WORD   MBR


                                ;       REWIND COMMAND PACKET (USED IN ERROR RECOVERY ONLY)

                        002330  .       =       .+3&177774      ;MUST BE ON A MODULE 4 BOUNDARY.
002330  102010          RWCPK::  .WORD   RWD
002332  000001                   .WORD   1


                                ;       WORK AREA FOR ANALYSIS OF MESSAGE PACKET CONTENTS.

002334  000007          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.
                                        ;
002352  000007                          MSGPK0:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #0
002370  000007                          MSGPK1:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #1
002406  000007                          MSGPK2:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #2
002424  000007                          MSGPK3:: .BLKW  7               ;MESSAGE PACKET FOR DEVICE #3
                                        ;
                                        ;       SET CHARACTERISTIC BLOCK.
                                        ;
002442  002352                          SCHBK:: MSGPK0                  ;1ST WORD:: MSGPKT ADDR LO(SET UP BY EXCUTE ROUT
002444  000000                                  0                       ;2ND WORD:: MSGPKT ADDR HI.
002446  000016                                  MSGCNT                  ;3RD WORD:: MSG BUFFER LENGTH (BYTES)
002450  000040                                  CH.EAI                  ;4TH WORD:: CHARACTERISTICS WORD(SET BY SETUP RO
                                        ;
                                        ;       TS04 REGISTER ADDRESSES.
                                        ;
002452  000004                          TSDB:: .BLKW    4               ;TS04 DATA BUFFER ADDRESSES.
002462  000004                          TSSR:: .BLKW    4               ;TS04 STATUS REGISTER ADDRESSES.
002472  000004                          TSVCT:: .BLKW   4               ;TS04 VECTOR ADDRESSES.
                                        ;
                                        ;       ADDRESSES OF MESSAGE PACKETS.
                                        ;
002502  002352                          MSGPKA:: MSGPK0                 ;DEVICE 0.
002504  002370                                  MSGPK1                  ;DEVICE 1.
002506  002406                                  MSGPK2                  ;DEVICE 2.
002510  002424                                  MSGPK3                  ;DEVICE 3.
                                        ;
                                        ;       ADDRESSES OF INTERRUPT HANDLING ROUTINES.
                                        ;
002512  006316                          TS4INT:: TS4IN0                 ;DEVICE 0.
002514  006324                                  TS4IN1                  ;DEVICE 1.
002516  006332                                  TS4IN2                  ;DEVICE 2.
002520  006340                                  TS4IN3                  ;DEVICE 3.
                                        ;
                                        ;       TS04 CODE LEVELS, WILL BE STORED AFTER SCH CMD IN BASIC FUNCTION TEST
                                        ;
002522  000000                          TS4CL:: 0                       ;DEVICE 0
002524  000000                                  0                       ;DEVICE 1
002526  000000                                  0                       ;DEVICE 2
002530  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.
                                        ;
```

```
2511  002532  177774                    DEVTBL:: .WORD  NINUSE
2512  002534  177774                             .WORD  NINUSE
2513  002536  177774                             .WORD  NINUSE
2514  002540  177774                             .WORD  NINUSE
2515  002542  177777                             .WORD  END
2516
2517
2518                                    ;        BAD TAPE TABLE POINTER: USED BY WRITE RETRY ROUTINE
2519                                    ;        'WRTY" TO LOG BAD TAPE SPOTS ON UNITS UNDER TEST
2520
2521  002544  002774                    BTADDR:: BT0
2522  002546  003046                             BT1
2523  002550  003120                             BT2
2524  002552  003172                             BT3
```

```
2525                                    ;       COUNTER AREA.
2526
2527            002554                  CNTBGN=.
2528  002554    000020                  WRBC::  .BLKW   20              ;BYTES WRITTEN.
2529  002614    000020                  RRBC::  .BLKW   20              ;BYTES READ REV.
2530  002654    000020                  RFBC::  .BLKW   20              ;BYTES READ FWD.
2531  002714    000004                  WRREC:: .BLKW   4               ;RECOVERABLE WRITE ERRORS.
2532  002724    000004                  WRUNR:: .BLKW   4               ;UNRECOVERABLE WRITE ERRORS.
2533  002734    000004                  RRREC:: .BLKW   4               ;RECOVERABLE READ REV ERRORS.
2534  002744    000004                  RRUNR:: .BLKW   4               ;UNRECOVERABLE READ REV ERRORS.
2535  002754    000004                  RFREC:: .BLKW   4               ;RECOVERABLE READ FWDERRORS.
2536  002764    000004                  RFUNR:: .BLKW   4               ;UNRECOVERABLE READ FWD ERRORS.
2537  002774    000025                  BT0::   .BLKW   21.             ;UNIT 0 BAT TAPE SPOTS LOG
2538  003046    000025                  BT1::   .BLKW   21.             ;UNIT 1 BAT TAPE SPOTS LOG
2539  003120    000025                  BT2::   .BLKW   21.             ;UNIT 2 BAT TAPE SPOTS LOG
2540  003172    000025                  BT3::   .BLKW   21.             ;UNIT 3 BAT TAPE SPOTS LOG
2541  003244    000004                  WRTYCT:: .BLKW  4               ;WRITE RETRY COUNTER
2542  003254    000004                  PASCNT:: .BLKW  4               ;PASS COUNT.
2543  003264    000004                  SCCNT:: .BLKW   4               ;SPECIAL CONDITION COUNT.
2544  003274    000004                  VFYCNT:: .BLKW  4               ;COUNT OF TS04 DATA COMPARE ERRORS.
2545  003304    000004                  HRDCNT:: .BLKW  4               ;COUNT OF HARD ERRORS.
2546  003314    000004                  FTLCNT:: .BLKW  4               ;COUNT OF FATAL ERRORS.
2547            003324                  CNTEND=.                        ;END OF STATICTICAL COUNTERS.
2548  003324    000004                  RECCNT:: .BLKW  4               ;NUMBER OF RECORDS FROM BOT: CLEARED  ON REWIND
2549                                                                    ;AND WHEN RESTARTING OR CONTINUING TEST 2.
2550
2551
2552                                    ;       THE FOLLOWING ARE THE DEFINITIONS OF VARIABLES
2553                                    ;       USED BY THE PROGRAM.
2554
2555  003334    000000                  DATAWT:: .WORD  0               ;WRITE BUFFER ADDRESS.
2556  003336    000000                  DATARD:: .WORD  0               ;READ BUFFER ADDRESS.
2557  003340    000000                  NCNT::  .WORD   0               ;STORAGE FOR VALUE OF N.
2558  003342    000000                  NCNT1:: .WORD   0               ;TEMP STORAGE FOR VALUE OF N.
2559  003344    000000                  BRFCNT:: .WORD  0               ;STORAGE FOR BPCR VALUE.
2560  003346    177777                  CMDWRD:: .WORD  END             ;CONTAINS COMMAND WORD BEING EXECUTED PRESENTLY.
2561  003350    177777                  CMDSAV:: .WORD  END             ;SAVE LOCATION FOR CMD WORD DURING ERROR RECOVER
2562  003352    177777                  PCMDWD:: .WORD  END             ;CONTAINS PREVOUS COMMAND WORD.
2563  003354    000000                  CMDLG:: .WORD   0               ;CURRENT COMMAND LOGGING CODE.
2564  003356    000000                  LENMSK:: .WORD  0               ;RANDOM WRITE LENGTH MASK, TO BE SET UP BY TESTS
2565  003360    153624                  RANB::  .WORD   153624          ;RANDOM # GENERATOR BASE.
2566  003362    032561                  RANS::  .WORD   32561           ;RANDOM # SAVE LOCATION.
2567  003364    000000                  TIME1:: .WORD   0               ;TIME COUNT 1.
2568  003366    000000                  TIME2:: .WORD   0               ;TIME COUNT 2.
2569  003370    000000                  JLOOP:: .WORD   0               ;JMP COMMAND LOOP COUNT.
2570  003372    000000                  JLOC::  .WORD   0               ;JMP COMMAND LOCATION COUNT.
2571  003374    000000                  PATERN:: .WORD  0               ;PATTERN SELECT CODE.
2572  003376    000000                  CTCC::  .WORD   0               ;CURRENT TERMINATION CLASS CODE.
2573  003400    000000                  R5SAVE:: .WORD  0               ;LOCATION FOR SAVING CURRENT DEVICE POINTER.
2574  003402    000000                  TSSREG:: .WORD  0               ;CURRENT STATUS REGISTER.
```

```
2575                                    ;       ERROR FLAG AREA, THESE FLAGS ARE CLEARED DURING INITIALIZATION AND
2576                                    ;       AFTER EACH COMMAND IS COMPLETED.
2577
2578            003404                  BGNFLG=.
2579  003404   000000                   RETRYC:: .WORD  0               ;# OF RECOVERY ATTEMPTS EXECUTED.
2580  003406      000                   RPTCNT:: .BYTE  0       ;WRITE REPEAT ON SAME SPOT CNTR: 4 PER WRITE RETRY
2581  003407      000                   WRTYFG:: .BYTE  0               ;WRITE RETRY ON SAME SPOT IN PROGRESS FLAG
2582  003410      000                   WRTYER:: .BYTE  0               ;WRITE RETRY ON SAME SPOT ERROR FLAG
2583  003411      000                   RECLOG:: .BYTE  0               ;RECORD COUNT HAS BEEN UPDATED FOR THIS RECORD.
2584  003412      000                   ERLOG:: .BYTE   0               ;DATA BYTES AND ERRORS HAVE BEEN LOGGED FOR THIS
2585  003413      000                   RWERR:: .BYTE   0               ;READ/WRITE ERROR HAS OCCURED.
2586  003414      000                   UNREC:: .BYTE   0               ;UNRECOVERABLE ERROR HAS OCCURED.
2587  003415      000                   ERRREC:: .BYTE  0               ;ERROR RECOVERY MODE.
2588                                            .EVEN
2589            003416                  ENDERF=.
2590
2591                                    ;       ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED DURING INITIALIZATION.
2592
2593  003416   000004                   INTFLG:: .BLKW  4               ;INTERRUPT OCCURRED FLAGS FOR EACH DEVICE.
2594  003426   000004                   EOTFLG:: .BLKW  4               ;EOT/BOT FLAGS FOR EACH DEVICE (XSTATO).
2595  003436   000000                   BTPT::   .WORD  0               ;BAD TAPE SPOT POINTER TO BTO BT3 VIA BTADDR
2596  003440      000                   EXPBOT:: .BYTE  0               ;BOT IS EXPECTED, DO NOT ABORT ON BOT/FUNC RTI.
2597  003441      000                   RANDOM:: .BYTE  0               ;RANDOM EVERYTHING FLAG
2598  003442      000                   VFYFLG:: .BYTE  0               ;SET DURING WRITE/VERIFY COMMAND.
2599  003443      000                   RPTFLG:: .BYTE  0               ;PERFORMANCE REPORT HAS BEEN REQUESTED.
2600  003444      000                   SWBFLG:: .BYTE  0               ;ENABLES SWAP BYTE FUNCTION WHEN NOT EQUAL TO ZE
2601  003445      000                   IRE::   .BYTE   0               ;INHIBIT RESIDUAL FRAME COUNT ERROR REPORT.
2602  003446      000                   DROPED:: .BYTE  0               ;CURRENT UNIT HAS BEEN DROPPED
2603  003447      000                   T1SWB::  .BYTE  0               ;TEST1 SWAP BYTES FLAG
2604  003450      000                   ALLEOT:: .BYTE  0               ;ALL UNITS @ EOT FLAG
2605  003451      000                   ERSFLG:: .BYTE  0               ;ERASE FLAG:  DO ERASE AFTER A SPACE REV TO DE-E
2606                                                                    ;BADLY WRITTEN RECORD.  1 TO 4 ERASES LEAVING
2607                                                                    ;A 3 TO 12 INCH GAP MAY RESULT.
2608                                            .EVEN
2609            003452                  ENDFLG=.
2610
2611                                    ;       ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED ONLY AFTER BEING CHECKED.
2612
2613  003452      000                   STAFLG:: .BYTE  0               ;START FLAG   SET BY INIT CODE IF STARTING.
2614  003453      000                   PWRFLG:: .BYTE  0               ;POWER FAILURE FLAG - SET ONLY DURING INIT.
2615  003454      000                   TRAPD4:: .BYTE  0               ;TRAPED AT 4 FLAG
2616  003455      000                   MISCFG:: .BYTE  0               ;MISCELLANEOUS FLAG
2617
2618                                    ;       OPERATOR FLAG SETTINGS PASSED BY DIAG. SUPERVISOR IN A 16 BIT WORD
2619                                    ;       SEE GLOBAL EQUATES SECTION FOR FLAG BIT LIST
2620
2621  003456   000000                   OPFLAG:: .WORD  0       ;READ ONLY OPERATOR FLAG WORD
2622                                            .EVEN
```

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

003460  140004              CMDSEQ:: .WORD  SCH             ;SET CHARACTERISTICS.
003462  000040                       .WORD  CH.EAI
003464  000001                       .WORD  1
003466  000000                       .WORD  0
003470  102010              CMDSE2:: .WORD  RWD             ;REWIND.
003472  000001                       .WORD  1               ;BYTE COUNT.
003474  000001                       .WORD  1               ;ONCE.
003476  000007                       .WORD  RANP            ;PATTERN.
003500  104005                       .WORD  WRT             ;WRITE.
003502  004000                       .WORD  DATCNT          ;MAX BUFFER LENGTH.
003504  076400                       .WORD  32000.          ;32,000 RECORDS.
003506  000007                       .WORD  RANP            ;RANDOM PATTERN.
003510  104401                       .WORD  RDR             ;READ REV.
003512  004000                       .WORD  DATCNT          ;MAX BUFFER LENGTH.
003514  076400                       .WORD  32000.          ;32,000 RECORDS
003516  000007                       .WORD  RANP            ;RANDOM PATTERN.
003520  104001                       .WORD  RDF             ;READ FWD.
003522  004000                       .WORD  DATCNT          ;MAX BUFFER LENGTH.
003524  076400                       .WORD  32000.          ;32,000 RECORDS.
003526  000007                       .WORD  RANP            ;RANDOM PATTERN.
003530  102010                       .WORD  RWD             ;REWIND.
003532  000001                       .WORD  1               ;BYTE COUNT.
003534  000001                       .WORD  1               ;ONCE.
003536  000007                       .WORD  RANP            ;PATTERN.
003540  000004                       .BLKW  4               ;EXTENSTION TO HOLD 1 MORE CMD.
003550  177777              SEQEND:: .WORD  END             ;SOFT END OF SEQUENCE TABLE.
003552  177777                       .WORD  END
003554  177777                       .WORD  END
003556  177777                       .WORD  END
003560  177777                       .WORD  END             ;HARD END OF SEQUENCE TABLE.
```

```
2657                                        ;THE FOLLOWING IS THE TS04 COMMAND TABLE
2658
2659  003562  100013          CMDTBL:: .WORD  DRI          ;DRIVE INIT.
2660  003564  104001                   .WORD  RDF          ;READ FORWARD.
2661  003566  104401                   .WORD  RDR          ;READ REVERSE.
2662  003570  104005                   .WORD  WRT          ;WRITE
2663  003572  104105                   .WORD  WTV          ;WRITE/VERIFY.  (WRITE ALL RECORDS, RDR AND
2664                                                       ;CHECK DATA ON ALL RECORDS, RDF AND
2665                                                       ;CHECK DATA ON ALL RECORDS.)
2666  003574  104010                   .WORD  SRF          ;SPACE "N" RECORDS FORWARD.
2667  003576  104410                   .WORD  SRR          ;SPACE "N" RECORDS REVERSE.
2668  003600  105401                   .WORD  RNR          ;READ NEXT REVERSE. I.E., SPACE FWD, READ REVERS
2669  003602  125401                   .WORD  RNF          ;READ NEXT FORWARD, I.E., READ FORWARD, SPACE RE
2670  003604  105001                   .WORD  RPF          ;READ PREVIOUS FORWARD. I.E., SPACE REVERSE, REA
2671  003606  125001                   .WORD  RPR          ;READ PREVIOUS REVERSE. I.E., READ REVERSE, SPAC
2672  003610  105005                   .WORD  WRR          ;WRITE RETRY.
2673  003612  102010                   .WORD  RWD          ;REWIND.
2674  003614  100012                   .WORD  MBR          ;MESSAGE BUFFER RELEASE
2675  003616  100011                   .WORD  WTM          ;WRITE TAPE MARK
2676  003620  101011                   .WORD  WTR          ;WRITE TAPE MARK RETRY.
2677  003622  105010                   .WORD  SFF          ;SPACE "N" FILES FORWARD.
2678  003624  105410                   .WORD  SFR          ;SPACE "N" FILES REVERSE.
2679  003626  100017                   .WORD  GES          ;GET EXTENDED STATUS.
2680  003630  100411                   .WORD  ERS          ;ERASE 3 INCHES OF TAPE.
2681  003632  100412                   .WORD  UNL          ;REWIND AND UNLOAD.
2682  003634  101012                   .WORD  CLN          ;CLEAR TAPE.
2683  003636  140004                   .WORD  SCH          ;SET CHARACTERISTICS.
2684  003640  100006                   .WORD  DIA          ;DIAGNOSTIC COMMAND.
2685  003642  000040                   .WORD  JMP          ;JUMP TO THE NTH COMMAND IN THE SEQUENCE.
2686  003644  000020                   .WORD  DLY          ;DELAY 'N' MS.
2687  003646  177777                   .WORD  END          ;END OF COMMAND TABLE
2688
```

```
2689                                    ;       THE FOLLOWING TABLE CONTAINS THE ASCII FOR EACH COMMAND.
2690
2691  003650  051104     111      CMDASC:: .ASCII /DRI/           ;DRIVE INIT.
2692  003653     122  043104              .ASCII  /RDF/           ;READ FORWARD.
2693  003656  042122     122              .ASCII  /RDR/           ;READ REVERSE.
2694  003661     127  052122              .ASCII  /WRT/           ;WRITE
2695  003664  052127     126              .ASCII  /WTV/           ;WRITE/VERIFY.  (WRITE ALL RECORDS, RDR AND CHEC
2696                                                              ;ON ALL RECORDS, RDF AND CHECK DATA ON ALL RECOR
2697  003667     123  043122              .ASCII  /SRF/           ;SPACE "N" RECORDS FORWARD.
2698  003672  051123     122              .ASCII  /SRR/           ;SPACE "N"  RECORDS REVERSE.
2699  003675     122  051116              .ASCII  /RNR/           ;READ NEXT REVERSE. I.E., SPACE FWD READ REVERSE
2700  003700  047122     106              .ASCII  /RNF/           ;READ NEXT FORWARD, I.E., READ FORWARD, SPACE RE
2701  003703     122  043120              .ASCII  /RPF/           ;READ PREVIOUS FORWARD. IE., SPACE REVERSE, READ
2702  003706  050122     122              .ASCII  /RPR/           ;READ PREVIOUS REVERSE. IE., READ REVERSE, SPACE
2703  003711     127  051122              .ASCII  /WRR/           ;WRITE RETRY.
2704  003714  053522     104              .ASCII  /RWD/           ;REWIND.
2705  003717     115  051102              .ASCII  /MBR/           ;MESSAGE BUFFER RELEASE
2706  003722  052127     115              .ASCII  /WTM/           ;WRITE TAPE MARK
2707  003725     127  051124              .ASCII  /WTR/           ;WRITE TAPE MARK RETRY.
2708  003730  043123     106              .ASCII  /SFF/           ;SPACE "N" FILES FORWARD.
2709  003733     123  051106              .ASCII  /SFR/           ;SPACE "N" FILES REVERSE.
2710  003736  042507     123              .ASCII  /GES/           ;GET EXTENDED STATUS.
2711  003741     105  051522              .ASCII  /ERS/           ;ERASE 3 INCHES OF TAPE.
2712  003744  047125     114              .ASCII  /UNL/           ;REWIND AND UNLOAD.
2713  003747     103  047114              .ASCII  /CLN/           ;CLEAN TAPE.
2714  003752  041523     110              .ASCII  /SCH/           ;SET CHARACTERISTICS.  WHERE BRF=200, 40, 20, 0.
2715                                                              ;SEE TS11/TS04 PROGRAMMING SPECIFICATION FOR DES
2716  003755     104  040511              .ASCII  /DIA/           ;DIAGNOSTICS.  SEE TS11/TS04 PROGRAMMING SPECIFI
2717                                                              ;FOR DESCRIPTION.  ODT MUST BE USED TO LOAD DIAG
2718                                                              ;INTO THE WRITE BUFFER BEFORE THIS CMD IS ISSUED
2719  003760  046512     120              .ASCII  /JMP/           ;JUMP TO THE NTH COMMAND IN THE COMMAND
2720                                                              ;SEQUENCE TABLE, WHERE N IS DEFINED IN
2721                                                              ;THE # OF OPERATIONS.
2722  003763     104  054514              .ASCII  /DLY/           ;DELAY  N  MS, WHERE N IS DEFINED IN
2723                                                              ;THE # OF OPERATIONS.
2724  003766  047105     104              .ASCII  /END/           ;END OF COMMAND SEQUENCE.
2725          003772                      .EVEN
2726
2727
2728
```

```
                        .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

003772  047045  040445  047125  CODELM:: .ASCIZ /%N%AUNIT %D1%A TS11 CODE LEVEL P%03%N%N/
                                .EVEN
004042  054130  020130  046503  HALTM:: .ASCIZ  /XXX CMD - TYPE <CR> TO CONTINUE/
004102  046503  020104  040520  CMDPKM:: .ASCIZ /CMD PACKET ADR NOT ON MODULO 4 BOUNDARY: RELOAD!/
        004164                  .EVEN
004164  040504  040524  041440  WTVERM:: .ASCIZ /DATA COMPARE ERROR/
004207     116  020117  051524  TOERM:: .ASCIZ  /NO TS11 RESPONSE/
004230  047125  042504  044506  SCERM:: .ASCIZ  /UNDEFINED SPEC COND/
004254  043122  020103  047516  RFCERM:: .ASCIZ /RFC NON ZERO/
004271     124  030523  020061  NSSRM:: .ASCIZ  /TS11 NOT READY/
004310  042522  051124  020131  RLEXM:: .ASCIZ  /RETRY LIMIT EXCEEDED/
004335     125  044516  020124  ATTNM:: .ASCIZ  /UNIT OFF LINE/
004353     106  047125  052103  FUNRM:: .ASCIZ  /FUNCTION REJECT/
004373     106  052101  046101  FATSM:: .ASCIZ  /FATAL SUBSYSTEM ERROR/
004421     116  020117  047111  NOINTM:: .ASCIZ /NO INTERRUPT/
004436  040524  042520  051440  TSAM::  .ASCIZ  /TAPE STATUS ALERT/
004460  047524  020117  040515  TOOMM:: .ASCIZ  /TOO MANY INTERRUPTS/
004504  040503  051520  040524  RNYM::  .ASCIZ  /CAPSTAN RUNAWAY GET STATUS RESULTS:/
004550  042522  047503  042526  RERM::  .ASCIZ  /RECOVERABLE ERROR/
004572  047125  042522  047503  URERM:: .ASCIZ  /UNRECOVERABLE ERROR/
004616  047045  040445  051104  DROPDM:: .ASCIZ /%N%ADROPPED UNIT %D1%N/
004645     045  022516  040501  AUDRPM:: .ASCIZ /%N%AALL UNITS DROPPED%N%N/
004677     045  022516  041101  DTAER2:: .ASCIZ "%N%ABYTE:%D4%S2%AWAS:%88%S2%AS/B:%88%N"
004746  042045  022464  020101  DTAER3:: .ASCIZ "%D4%A BYTES IN ERROR OUT OF %D4%N"
005010  040445  047516  042040  DTAER4:: .ASCIZ /%ANO DATA READ%N/
005031     045  051101  041505  DTAER5:: .ASCIZ /%ARECORD TOO LONG: >%04%A BYTES%N/
005073     045  051101  041505  NURTY1:: .ASCIZ /%ARECOVERED ON RETRY #%D2%N
005127     045  052501  044516  OFLINM:: .ASCIZ /%AUNIT %D1%A OFF LINE%N/
005157     045  043501  052105  GETSTM:: .ASCIZ /%AGET STATUS CMD RESULTS:%N/
005213     045  000116          CRLF::  .ASCIZ  /%N/
005216  047045  051445  000067  CRLFSP:: .ASCIZ /%N%S7/
                                .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..
                                        ;

005224                                          BGNMSG  DTAERM
005224                                  DTAERM::
005224                                          PRINTB  #STAER1,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
005224  016546  003324                                                          MOV     RECCNT(R
005230  016546  003254                                                          MOV     PASCNT(R
005234  016546  002532                                                          MOV     DEVTBL(R
005240  012746  005704                                                          MOV     #STAER1,
005244  012746  000004                                                          MOV     #4,-(SP)
005250  010600                                                                  MOV     SP,R0
005252  104414                                                                  TRAP    C$PNTB
005254  062706  000012                                                          ADD     #12,SP
005260                                          PRINTB  #STAER7
005260  012746  005776                                                          MOV     #STAER7,
005264  012746  000001                                                          MOV     #1,-(SP)
005270  010600                                                                  MOV     SP,R0
005272  104414                                                                  TRAP    C$PNTB
005274  062706  000004                                                          ADD     #4,SP
005300                                          LET RECRED :* R2                ;SAVE R2
005300  010237  006312                                                          MOV     R2,RECRE
005304                                          LET TIME1 :* R3                 ;SAVE R3
005304  010337  003364                                                          MOV     R3,TIME1
005310                                          LET TIME2 :* R4                 ;SAVE R4
005310  010437  003366                                                          MOV     R4,TIME2
005314  004737  006346                                  JSR PC,RECTAP           ;RETRIEVE RECORD READ
005320                                          LET R2 :* RECRED                ;RESTORE R2
005320  013702  006312                                                          MOV     RECRED,R
005324                                          LET RECRED :* R3                ;SAVE RECORD READ
005324  010337  006312                                                          MOV     R3,RECRE
005330                                          LET R3 :* TIME1                 ;RESTORE R3
005330  013703  003364                                                          MOV     TIME1,R3
005334                                          LET R4 :* TIME2                 ;RESTORE R4
005334  013704  003366                                                          MOV     TIME2,R4
005340                                          PRINTB  #STAER6,RECRED          ;PRINT RECORD READ
005340  013746  006312                                                          MOV     RECRED,
005344  012746  006026                                                          MOV     #STAER6,
005350  012746  000002                                                          MOV     #2,-(SP)
005354  010600                                                                  MOV     SP,R0
005356  104414                                                                  TRAP    C$PNTB
005360  062706  000006                                                          ADD     #6,SP
005364                                          EXIT    MSG
005364  000167                                                                  .WORD   .$JMP
005366  000000                                                                  .WORD   L10002-.
                                        .EVEN

005370                                          ENDMSG
005370                                  L10002:
005370  104423                                                                  TRAP    C$MSG
```

```
2800
2801  005372                                  BGNMSG  STAERM
2802  005372                          STAERM::
2803  005372                                  PRINTB  #STAER1,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
2804  005372  016546  003324                                                  MOV     RECCNT(R
2805  005376  016546  003254                                                  MOV     PASCNT(R
2806  005402  016546  002532                                                  MOV     DEVTBL(R
2807  005406  012746  005704                                                  MOV     #STAER1,
2808  005412  012746  000004                                                  MOV     #4.,(SP)
2809  005416  010600                                                          MOV     SP,R0
2810  005420  104414                                                          TRAP    C$PNTB
2811  005422  062706  000012                                                  ADD     #12.,SP
2812  005426                                  PRINTB  #STAER7
2813  005426  012746  005776                                                  MOV     #STAER7,
2814  005432  012746  000001                                                  MOV     #1.,(SP)
2815  005436  010600                                                          MOV     SP,R0
2816  005440  104414                                                          TRAP    C$PNTB
2817  005442  062706  000004                                                  ADD     #4.,SP
2818  005446                                  LET R2 := CMDPKT CLR.BY #177740
2819  005446  013702  002310                                                  MOV     CMDPKT,R
2820  005452  042702  177740                                                  BIC     #177740,
2821  005456                                  LET R2 := R2   #1
2822  005456  005302                                                          DEC     R2
2823  005460                                  IF R2 EQ #0 THEN                ;IF CMD IS A READ
2824  005460  005702                                                          TST     R2
2825  005462  001016                                                          BNE     50000$
2826  005464  004737  006346                    JSR PC,RECTAP                 ;THEN RETRIEVE
2827  005470                                    LET RECRED := R3              ;AND
2828  005470  010337  006312                                                  MOV     R3,RECRE
2829  005474                                    PRINTB #STAER6,RECRED         ;TYPE RECORD READ
2830  005474  013746  006312                                                  MOV     RECRED,-
2831  005500  012746  006026                                                  MOV     #STAER6,
2832  005504  012746  000002                                                  MOV     #2.,-(SP)
2833  005510  010600                                                          MOV     SP,R0
2834  005512  104414                                                          TRAP    C$PNTB
2835  005514  062706  000006                                                  ADD     #6,SP
2836  005520                                  ENDIF
2837  005520                                                          50000$:
2838  005520                                  PRINTX  #STAER2
2839  005520  012746  006062                                                  MOV     #STAER2,
2840  005524  012746  000001                                                  MOV     #1.,-(SP)
2841  005530  010600                                                          MOV     SP,R0
2842  005532  104415                                                          TRAP    C$PNTX
2843  005534  062706  000004                                                  ADD     #4,SP
2844  005540                                  PRINTX  #STAER3,CMDPKT,@TSDB(R5),MSGPKT+MS.RFC,TSSREG,CTCC
2845  005540  013746  003376                                                  MOV     CTCC,(S
2846  005544  013746  003402                                                  MOV     TSSREG,-
2847  005550  013746  002340                                                  MOV     MSGPKT+M
2848  005554  017546  002452                                                  MOV     @TSDB(R5
2849  005560  013746  002310                                                  MOV     CMDPKT,
2850  005564  012746  006141                                                  MOV     #STAER3,
2851  005570  012746  000006                                                  MOV     #6.,(SP
2852  005574  010600                                                          MOV     SP,R0
2853  005576  104415                                                          TRAP    C$PNTX
2854  005600  062706  000016                                                  ADD     #16,SP
2855  005604                                  PRINTX  #STAER4,CMDPKT+2,CMDPKT+4,CMDPKT+6
```

```
2856  005604  013746  002316                                                  MOV     CMDPKT+6
2857  005610  013746  002314                                                  MOV     CMDPKT+4
2858  005614  013746  002312                                                  MOV     CMDPKT+2
2859  005620  012746  006177                                                  MOV     #STAER4,
2860  005624  012746  000004                                                  MOV     #4, (SP)
2861  005630  010600                                                          MOV     SP,R0
2862  005632  104415                                                          TRAP    C$PNTX
2863  005634  062706  000012                                                  ADD     #12,SP
2864  005640                                PRINTX  #STAER5,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3
2865  005640  013746  002350                                                  MOV     MSGPKT+M
2866  005644  013746  002346                                                  MOV     MSGPKT+M
2867  005650  013746  002344                                                  MOV     MSGPKT+M
2868  005654  013746  002342                                                  MOV     MSGPKT+M
2869  005660  012746  006217                                                  MOV     #STAER5,
2870  005664  012746  000005                                                  MOV     #5,-(SP)
2871  005670  010600                                                          MOV     SP,R0
2872  005672  104415                                                          TRAP    C$PNTX
2873  005674  062706  000014                                                  ADD     #14,SP
2874  005700                                EXIT    MSG
2875  005700  000167                                                          .WORD   J$JMP
2876  005702  000410                                                          .WORD   L10003-2
2877
2878                                        .NLIST  BEX
      005704  040445  054130  020130  STAER1: .ASCIZ  /%AXXX CMD FAILED   UNIT %D1%S3%APASS:%D5%S3%ARECORD:%D5%N/
                                        .EVEN
      005776  040445  051120  053105  STAER7: .ASCIZ  /%APREVIOUS CMD WAS  XXX/
      006026  051445  030461  040445  STAER6: .ASCIZ  /%S11%A. RECORD READ:%D5%A ./
      006062  047045  040445  046503  STAER2: .ASCIZ  /%N%ACMDPKT%S2%ATSBA%S4%ARFC%S5%ATSSR%S3%ATCC%N/
      006141     045  033117  051445  STAER3: .ASCIZ  /%06%S2%06%S2%06%S2%06%S2%D1%N/
      006177     045  033117  047045  STAER4: .ASCII  /%06%N/
      006204  047445  022466     116          .ASCII  /%06%N/
      006211     045  033117  047045          .ASCIZ  /%06%N/
      006217     045  054101  052123  STAER5: .ASCII  /%AXST0%S4%AXST1%S4%AXST2%S4%AXST3%N'
      006262  047445  022466  031123          .ASCIZ  /%06%S2%06%S2%06%S2%06%N/
                                        .LIST   BEX
2879                                    .EVEN
2880  006312  000000  RECRED: .WORD   0                       ;RECORD READ FROM TAPE
2881
2882  006314                            ENDMSG
2883  006314                L10003:
2884  006314  104423                                                          TRAP    C$MSG
```

```
                                        .SBTTL  GLOBAL SUBROUTINES SECTION

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

                                        ;       MODULES TO HANDLE TS04 INTERRUPTS.

006316                                          BGNSRV  TS4IN0                  ;DEVICE 0.
006316                                  TS4IN0::
006316                                          LET INTFLG := INTFLG + #1       ;SET INTERRUPT OCCURRED FLAG.
006316  005237  003416                                                          INC     INTFLG
006322                                          ENDSRV
006322                                  L10004:
006322  000002                                                                  RTI

006324                                          BGNSRV  TS4IN1                  ;DEVICE 1.
006324                                  TS4IN1::
006324                                          LET INTFLG+2 := INTFLG+2 + #1   ;SET INTERRUPT OCCURRED FLAG.
006324  005237  003420                                                          INC     INTFLG+2
006330                                          ENDSRV
006330                                  L10005:
006330  000002                                                                  RTI

006332                                          BGNSRV  TS4IN2                  ;DEVICE 2.
006332                                  TS4IN2::
006332                                          LET INTFLG+4 := INTFLG+4 + #1   ;SET INTERRUPT OCCURRED FLAG.
006332  005237  003422                                                          INC     INTFLG+4
006336                                          ENDSRV
006336                                  L10006:
006336  000002                                                                  RTI

006340                                          BGNSRV  TS4IN3                  ;DEVICE 3.
006340                                  TS4IN3::
006340                                          LET INTFLG+6 := INTFLG+6 + #1   ;SET INTERRUPT OCCURRED FLAG.
006340  005237  003424                                                          INC     INTFLG+6
006344                                          ENDSRV
006344                                  L10007:
006344  000002                                                                  RTI
```

```
                                        ;       SUBROUTINE TO RETRIEVE RECORD COUNT READ FROM TAPE FOR ERROR
                                        ;       PRINTS.
                                        ;       INPUTS:
                                        ;       OUTPUTS:  R3 = RECORD COUNT READ
                                        ;       REGISTERS:  R2, R3, R4
                                        ;       CALLS:

006346                                  RECTAP::IF @MOD.C0 SETIN CMDWRD THEN    ;READ REV FETCH
006346  032737  000400  003346                                                          BIT     @MOD.C0,
006354  001430                                                                         BEQ     50001$
006356                                    LET R2 := MSGPKT+MS.RFC + DATARD ;FIND LAST READ AD.
006356  013702  002340                                                                 MOV     MSGPKT+M
006362  063702  003336                                                                 ADD     DATARD,R
006366                                    IF @BIT00 SETIN R2 THEN       ;ODD AD., REASSEMBLE
006366  032702  000001                                                                 BIT     @BIT00,P
006372  001417                                                                         BEQ     50002$
006374                                      LET R2 := R2 + @1           ;REC COUNT STARTING
006374  005202                                                                         INC     R2
006376                                      LET R3 :B= (R2) CLR.BY @177400  ;WITH UPPER BYTE FETCH
006376  111203                                                                         MOVB    (R2),R3
006400  142703  177400                                                                 BICB    @177400,
006404                                      LET R3 := SWAP R3           ;
006404  000303                                                                         SWAB    R3
006406                                      LET R2 := R2   @1           ;LOWER BYTE AD.
006406  005302                                                                         DEC     R2
006410                                      IFB SWBFLG NE @0 THEN
006410  105737  003444                                                                 TSTB    SWBFLG
006414  001401                                                                         BEQ     50003$
006416                                        LET R2 := R2 - @1         ;LOWER BYTE AD. ON SWAP
006416  005302                                                                         DEC     R2
006420                                      ENDIF
006420                                      LET R4 :B= (R2) CLR.BY @177400 ;FETCH LOWER BYTE
006420  111204                                                          50003$:  MOVB    (R2),R4
006422  142704  177400                                                          BICB    @177400,
006426                                      LET R3 := R3 OR R4          ;MERGE BYTES
006426  050403                                                                 BIS     R4,R3
006430                                    ELSE
006430  000401                                                                 BR      50004$
006432                                                                  50002$:
006432                                      LET R3 := (R2)              ;EVEN AD. FETCH
006432  011203                                                                 MOV     (R2),R3
006434                                    ENDIF
006434                                                                  50004$:
006434                                  ELSE
006434  000402                                                                 BR      50005$
006436                                                                  50001$:
006436                                    LET R3 := @DATARD             ;READ FWD FETCH
006436  017703  174674                                                          MOV     @DATARD,
006442                                  ENDIF
006442                                                                  50005$:

006442  000207                          RTS     PC
```

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

006444                  SETCH:: LET R1 := #CMDSEQ               ;INIT COMMAND SEQUENCE TABLE POINTER.
006444  012701  003460                                                  MOV     #CMDSEQ,
006450  012721  140004          MOV     #SCH,(R1)+              ;THIS CODE SETS UP A SET CHARACTERISTIC
006454  012721  000040          MOV     #DFTSCH,(R1)+           ;COMMAND AS THE FIRST COMMAND IN THE
006460  012721  000001          MOV     #1,(R1)+                ;SEQUENCE TABLE.
006464  005721                  TST     (R1)+                   ;SKIP PATTERN LOCATION.
006466  000207                  RTS PC


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

006470                  SETRW:: LET (R1)+ := #RWD               ;CMD = REWIND.
006470  012721  102010                                                  MOV     #RWD,(R1
006474                          LET (R1)+ := #1                 ;BRF.
006474  012721  000001                                                  MOV     #1,(R1)+
006500                          LET (R1)+ := #1                 ;# OF OPERATIONS.
006500  012721  000001                                                  MOV     #1,(R1)+
006504  005721                  TST (R1)+                       ;SKIP PATTERN.
006506  000207                  RTS     PC                      ;RETURN
```

```
3010                                    ;       SUBROUTINE TO EXECUTE ALL COMMANDS IN THE SEQUENCE TABLE ON ALL
3011                                    ;       DEVICES.
3012                                    ;       INPUTS:
3013                                    ;       OUTPUTS:        R2 = TERMINATION INDICATOR (0-END OF TABLE,1-EOT)
3014                                    ;       REGISTERS:
3015                                    ;       CALLS:          CMDAC,SETUP,EXSUB,CKHAE,NEXTU,FIRSTU,VFYDAT.
3016
3017  006510                            EXALL:: LET R1 := @CMDSEQ               ;INIT SEQUENCE TABLE POINTER.
3018  006510  012701  003460                                                            MOV     @CMDSEQ,
3019  006514                                    WHILE (R1) NE @END DO             ;WHILE THERE ARE CMDS IN THE SEQUENCE TA
3020  006514                                                                                50006$:
3021  006514  021127  177777                                                            CMP     (R1),@EN
3022  006520  001527                                                                    BEQ     50007$
3023  006522  004737  007452                      JSR   PC,SETUP                ;GO SETUP THE COMMAND BLOCK.
3024  006526                                      WHILE NCNT LT NCNT1 DO        ;WHILE THERE ARE RECORDS REMAINING:
3025  006526                                                                                50010$:
3026  006526  023737  003340  003342                                                      CMP     NCNT,NCN
3027  006534  002116                                                                    BGE     50011$
3028  006536  004737  007344                        JSR PC,CMDAC                ;STORE CMD ASCII IN ERROR MESSAGE.
3029  006542                                        IFB RANDOM NE @0 THEN       ;IF IN RANDOM MODE:
3030  006542  105737  003441                                                            TSTB    RANDOM
3031  006546  001435                                                                    BEQ     50012$
3032  006550                                          IF CMDWRD EQ @WRT THEN    ;IF CMD IS A WRITE THEN:
3033  006550  023727  003346  104005                                                      CMP     CMDWRD,@
3034  006556  001031                                                                    BNE     50013$
3035  006560                                            IFB VFYFLG EQ @0 THEN   ;IF DATA IS NOT TO BE VERIFIED THEN:
3036  006560  105737  003442                                                            TSTB    VFYFLG
3037  006564  001026                                                                    BNE     50014$
3038  006566                                              LET RANB := RANB + RANS  ;GENERATE
3039  006566  063737  003362  003360                                                      ADD     RANS,RAN
3040  006574                                              LET RANS := RANS + RANB  ;RANDOM
3041  006574  063737  003360  003362                                                      ADD     RANB,RAN
3042  006602                                            LET BRFCNT := RANS        ;LENGTH
3043  006602  013737  003362  003344                                                      MOV     RANS,BRF
3044  006610                                            LET BRFCNT := BRFCNT CLR.BY LENMSK  ;MASK RANDOM LENGTH.
3045  006610  043737  003356  003344                                                      BIC     LENMSK,B
3046  006616                                            IF BRFCNT LT @18. THEN        ;DO NOT ALLOW BYTE COUNT OF LESS
3047  006616  023727  003344  000022                                                      CMP     BRFCNT,@
3048  006624  002003                                                                    BGE     50015$
3049  006626                                              LET BRFCNT := @18.  ;CHANGE COUNT OF 0 17 TO 18.
3050  006626  012737  000022  003344                                                      MOV     @18.,BRF
3051  006634                                            ENDIF
3052  006634                                                                                50015$:
3053  006634                                            LET CMDPKT+CP.CNT := BRFCNT  ;MOVE BRF TO CMD PACKET.
3054  006634  013737  003344  002316                                                      MOV     BRFCNT,C
3055  006642                                          ENDIF
3056  006642                                                                                50014$:
3057  006642                                        ENDIF
3058  006642                                                                                50013$:
3059  006642                                    ENDIF
3060  006642
3061  006642  004737  007004                    JSR PC,EXSUB                  ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STAT
3062  006646  004737  016060                    JSR PC,CKHAE                  ;CHECK HALT AFTER EACH CMD FLAG.
3063  006652                                    LET R2 := @1                  ;SET ALL UNITS AT BOT/EOT.
3064  006652  012702  000001                                                            MOV     @1,R2
3065  006656  004737  015452                    JSR PC,FIRSTU                 ;FIND FIRST UNIT.
```

```
3066  006662                                                  WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE UNITS:
3067  006662                                                                                  50016$:
3068  006662  026527  002532  177777                                                          CMP     DEVTBL(R
3069  006670  001426                                                                        BEQ     50017$
3070  006672                                                    IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
3071  006672  032737  000400  003346                                                        BIT     #MOD.CO,
3072  006700  001406                                                                        BEQ     50020$
3073  006702                                                      IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT THEN:
3074  006702  032765  000002  003426                                                        BIT     #XO.BOT,
3075  006710  001001                                                                        BNE     50021$
3076  006712                                                        LET R2 :- #0                    ;CLEAR EOT/BOT FLAG.
3077  006712  005002                                                                        CLR     R2
3078  006714                                                      ENDIF
3079  006714                                                                                  50021$:
3080  006714                                                    ELSE                        ;ELSE IF CMD IS NOT REVERSE:
3081  006714  000411                                                                        BR      50022$
3082  006716                                                                                  50020$:
3083  006716                                                      IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
3084  006716  032765  000001  003426                                                        BIT     #XO.EOT,
3085  006724  001404                                                                        BEQ     50023$
3086  006726  032737  000001  003346                                                        BIT     #CMD.CO,
3087  006734  001001                                                                        BNE     50024$
3088  006736                                                                                  50023$:
3089                                                                                  ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
3090  006736                                                        LET R2 :- #0              ;CLEAR EOT/BOT FLAG.
3091  006736  005002                                                                        CLR     R2
3092  006740                                                      ENDIF
3093  006740                                                                                  50024$:
3094  006740                                                    ENDIF
3095  006740                                                                                  50022$:
3096  006740  004737  015520                                   JSR PC,NEXTU                ;FIND NEXT UNIT
3097  006744                                                    ENDDO                       ;
3098  006744  000746                                                                        BR      50016$
3099  006746                                                                                  50017$:
3100  006746                                                    IF R2 EQ #1 THEN            ;IF ALL UNIT ARE AT EOT/BOT THEN:
3101  006746  020227  000001                                                                CMP     R2,#1
3102  006752  001001                                                                        BNE     50025$
3103  006754  000412                                             BR        EXARTN          ;RETURN WITH R2 = #1.
3104  006756                                                    ENDIF
3105  006756                                                                                  50025$:
3106  006756                                                    LET NCNT :- NCNT + #1       ;UPDATE RECORD COUNT.
3107  006756  005237  003340                                                                INC     NCNT
3108  006762                                                    LET PCMDWD := CMDWRD        ;SAVE PREVIOUS COMMAND WORD.
3109  006762  013737  003346  003352                                                        MOV     CMDWRD,P
3110  006770                                                  ENDDO
3111  006770  000656                                                                        BR      50010$
3112  006772                                                                                  50011$:
3113  006772  004737  014402                                   JSR   PC,VFYDAT             ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
3114                                                                                    ;VERIFY THE LAST N RECORDS OF DATA.
3115  006776                                                  ENDDO
3116  006776  000646                                                                        BR      50006$
3117  007000                                                                                  50007$:
3118  007000                                                  LET R2 := #0                ;SET NORMAL RETURN INDICATOR.
3119  007000  005002                                                                        CLR     R2
3120  007002  000207                                 EXARTN: RTS PC                      ;RETURN.
3121
```

```
3122
3123
3124                                    ;       SUBROUTINE TO ISSUE COMMAND TO ALL DEVICES, WAIT FOR
3125                                    ;       AL  INTERRUPTS, AND CHECK ALL STATUS.
3126                                    ;       INPUTS:
3127                                    ;       OUTPUTS:
3128                                    ;       REGISTERS:
3129                                    ;       CALLS:          EXCUTE,GOWAIT,NEXTU,FIRSTU.
3130
3131  007004  004737  015452            EXSUB::     JSR PC,FIRSTU               ;SET UP FOR FIRST UNIT.
3132  007010                              WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
3133  007010                                                                      50026$:
3134  007010  026527  002532  177777                                                CMP     DEVTBL(R
3135  007016  001465                                                               BEQ     50027$
3136  007020                                IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
3137  007020  032737  000400  003346                                                BIT     #MOD.CO,
3138  007026  001421                                                               BEQ     50030$
3139  007030                                  IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
3140  007030  032765  000002  003426                                                BIT     #XO.BOT,
3141  007036  001014                                                               BNE     50031$
3142  007040                                    IF #XO.EOT SETIN EOTFLG(R5) THEN  ;BUT IF AT EOT
3143  007040  032765  000001  003426                                                BIT     #XO.EOT,
3144  007046  001406                                                               BEQ     50032$
3145  007050                                      IFB ALLEOT NE #0 THEN       ;AND ALL OTHER  AT EOT
3146  007050  105737  003450                                                      TSTB    ALLEOT
3147  007054  001402                                                             BEQ     50033$
3148  007056  004737  010326                       JSR PC,EXCUTE               ;THEN EXECUTE REV CMD
3149  007062                                      ENDIF                         ;IF NOT ALL AT EOT, FREEZE UNIT(
3150  007062                                                                    50033$:
3151  007062                                    ELSE                            ;IF NOT AT BOT AND
3152  007062  000402                                                            BR      50034$
3153  007064                                                                    50032$:
3154  007064  004737  010326                     JSR PC,EXCUTE                 ;NOT AT EOT, EXEC REV CM
3155  007070                                    ENDIF
3156  007070                                                                    50034$:
3157  007070                                  ENDIF
3158  007070                                                                    50031$:
3159  007070                              ELSE                        ;ELSE IF CMD IS NOT REVERSE:
3160  007070  000435                                                            BR      50035$
3161  007072                                                                    50030$:
3162  007072                                IF CMDLG EQ #2 AND #XO.BOT SETIN EOTFLG(R5) THEN
3163  007072  023727  003354  000002                                                CMP     CMDLG,#2
3164  007100  001011                                                               BNE     50036$
3165  007102  032765  000002  003426                                                BIT     #XO.BOT,
3166  007110  001405                                                               BEQ     50036$
3167                                                          ;CLEAR BAD SPOT COUNTS WHEN WRITING FROM
3168  007112                                  LET BTPT := BTADDR(R5)
3169  007112  016537  002544  003436                                                MOV     BTADDR(R
3170  007120                                  LET @BTPT := #0
3171  007120  005077  174312                                                       CLR     @BTPT
3172  007124                                ENDIF
3173  007124                                                                    50036$:
3174  007124                                IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
3175  007124  032765  000001  003426                                                BIT     #XO.EOT,
3176  007132  001404                                                               BEQ     50037$
3177  007134  032737  000001  003346                                                BIT     #CMD.CO,
```

```
3178  007142  001003                                                                    BNE     50040$
3179  007144                                                                          50037$:
3180                                                                          ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
3181  007144  004737  010326                    JSR PC,EXCUTE                 ;ISSUE CMD TO TS04
3182  007150                                  ELSE
3183  007150  000405                                                                  BR      50041$
3184  007152                                                                        50040$:
3185  007152                                    IFB ALLEOT NE #0 THEN
3186  007152  105737  003450                                                          TSTB    ALLEOT
3187  007156  001402                                                                  BEQ     50042$
3188  007160  004737  010326                      JSR PC,EXCUTE               ;
3189  007164                                    ENDIF
3190  007164                                                                        50042$:
3191  007164                                    ENDIF
3192  007164                                                                        50041$:
3193  007164                                  ENDIF
3194  007164                                                                        50035$:
3195  007164  004737  015520                  JSR       PC,NEXTU            ;FIND NEXT UNIT IN TEST CYCLE.
3196  007170                                ENDDO
3197  007170  000707                                                                  BR      50026$
3198  007172                                                                        50027$:
3199  007172                                IFB RPTFLG NE #0 THEN               ;IF REPORT HAS BEEN REQUESTED THEN:
3200  007172  105737  003443                                                          TSTB    RPTFLG
3201  007176  001403                                                                  BEQ     50043$
3202  007200                                  LET RPTFLG :B  #0                   ;CLR THE FLAG.
3203  007200  105037  003443                                                          CLRB    RPTFLG
3204  007204                                  DORPT                               ;PRINT THE PERFORMANCE REPORT.
3205  007204  104424                                                                  TRAP    C$DRPT
3206  007206                                ENDIF                                 ;
3207  007206                                                                        50043$:
3208  007206  004737  015452                JSR PC,FIRSTU                       ;SET UP FOR FIRST UNIT.
3209  007212                                WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
3210  007212                                                                        50044$:
3211  007212  026527  002532  177777                                                  CMP     DEVTBL(R
3212  007220  001450                                                                  BEQ     50045$
3213  007222                                  IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
3214  007222  032737  000400  003346                                                  BIT     #MOD.CO.
3215  007230  001421                                                                  BEQ     50046$
3216  007232                                    IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
3217  007232  032765  000002  003426                                                  BIT     #XO.BOT.
3218  007240  001014                                                                  BNE     50047$
3219  007242                                      IF #XO.EOT SETIN EOTFLG(R5) THEN      ;BUT IF AT EOT
3220  007242  032765  000001  003426                                                  BIT     #XO.EOT.
3221  007250  001406                                                                  BEQ     50050$
3222  007252                                        IFB ALLEOT NE #0 THEN             ;AND ALL OTHERS AT EOT
3223  007252  105737  003450                                                          TSTB    ALLEOT
3224  007256  001402                                                                  BEQ     50051$
3225  007260  004737  010636                        JSR PC,GOWAIT                 ;THEN WAIT FOR CMD END
3226  007264                                      ENDIF                           ;IF NOT ALL AT EOT, DO N
3227  007264                                                                      50051$:
3228  007264                                    ELSE                              ;NOT AT BOT, AND NOT AT
3229  007264  000402                                                                  BR      50052$
3230  007266                                                                      50050$:
3231  007266  004737  010636                      JSR PC,GOWAIT                         ;WAIT FOR INT,CH
3232  007272                                    ENDIF
3233  007272                                                                      50052$:
```

```
3234  007272                                                    ENDIF
3235  007272                                                                                    50047$:
3236  007272                                                  ELSE                              ;ELSE IF CMD IS FORWARD:
3237  007272  000420                                                                                    BR      50053$
3238  007274                                                                                    50046$:
3239  007274                                                    IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
3240  007274  032765  000001  003426                                                                    BIT     #XO.EOT,
3241  007302  001404                                                                                    BEQ     50054$
3242  007304  032737  000001  003346                                                                    BIT     #CMD.CO,
3243  007312  001003                                                                                    BNE     50055$
3244  007314                                                                                    50054$:
3245                                                                                            ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
3246  007314  004737  010636                                      JSR PC,GOWAIT                   ;WAIT FOR INT,CHECK STATUS.
3247  007320                                                    ELSE
3248  007320  000405                                                                                    BR      50056$
3249  007322                                                                                    50055$:
3250  007322                                                      IFB ALLEOT NE #0 THEN
3251  007322  105737  003450                                                                          TSTB    ALLEOT
3252  007326  001402                                                                                    BEQ     50057$
3253  007330  004737  010636                                        JSR PC,GOWAIT
3254  007334                                                      ENDIF
3255  007334                                                                                    50057$:
3256  007334                                                    ENDIF
3257  007334                                                                                    50056$:
3258  007334                                                  ENDIF
3259  007334                                                                                    50053$:
3260  007334  004737  015520                                  JSR       PC,NEXTU              ;FIND NEXT UNIT IN TEST CYCLE.
3261  007340                                                ENDDO
3262  007340  000724                                                                                    BR      50044$
3263  007342                                                                                    50045$:
3264  007342  000207                                        RTS PC                            ;RETURN.
```

```
                                ;       THIS SUBROUTINE STORES THE ASCII FOR THE CURRENT COMMAND AND PREVIOUS
                                ;       COMMAND IN THE STANDARD ERROR MESSAGE.  ON ENTRY LOCATION CMDWRD
                                ;       CONTAINS CURRENT CMD AND LOCATION PCMDWD CONTAINS PREVIOUS CMD.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R3, R4.
                                ;       CALLS:          GCMDA
                                ;
007344                          CMDAC:: LET R4 := CMDWRD                ;R4 = CMD BINARY.
007344  013704  003346                                                          MOV     CMDWRD,R
007350  004737  007416                  JSR     PC,GCMDA                ;GET CMD ASCII.
007354  112337  005706                  MOVB    (R3)+,STAER1+2          ;MOVE CMD ASCII
007360  112337  005707                  MOVB    (R3)+,STAER1+3          ;
007364  111337  005710                  MOVB    (R3),STAER1+4           ;INTO MSG.
007370                                  LET R4 := PCMDWD                ;R4 = PREVIOUS CMD BINARY.
007370  013704  003352                                                          MOV     PCMDWD,R
007374  004737  007416                  JSR     PC,GCMDA                ;GET CMD ASCII.
007400                                  LET STAER7+24 :B= (R3)+         ;MOVE CMD ASCII
007400  112337  006022                                                          MOVB    (R3)+,ST
007404                                  LET STAER7+25 :B= (R3)+         ;
007404  112337  006023                                                          MOVB    (R3)+,ST
007410                                  LET STAER7+26 :B= (R3)          ;INTO MSG.
007410  111337  006024                                                          MOVB    (R3),STA
007414  000207                          RTS     PC                      ;RETURN. GO EXECUTE NEXT FUNCTION.


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

007416                          GCMDA:: LET R3 := 0                     ;INIT CMD TBL POINTER.
007416  005003                                                          CLR     R3
007420                                  WHILE CMDTBL(R3) NE R4 DO       ;UNTIL CURRENT CMD IS FOUND:
007420                                                                  50060$:
007420  026304  003562                                                          CMP     CMDTBL(R
007424  001403                                                          BEQ     50061$
007426                                    LET R3 := R3 + 2              ;SEARCH CMD TABLE.
007426  062703  000002                                                          ADD     #2,R3
007432                                  ENDDO
007432  000772                                                          BR      50060$
007434                                                                  50061$:
007434                                  LET R4 := R3
007434  010304                                                          MOV     R3,R4
007436                                  LET R3 := R3 SHIFT  1           ;POINT TO ASCII FOR THAT COMMAND
007436  006203                                                          ASR     R3
007440  000240                          NOP
007442  060403                          ADD     R4,R3
007444  062703  003650                  ADD     #CMDASC,R3
007450  000207                          RTS     PC                      ;RETURN.
```

```
3318                                    ;       THIS SUBROUTINE LOADS THE TSO4 COMMAND PACKET FROM ONE
3319                                    ;       ENTRY IN THE SEQUENCE TABLE.
3320                                    ;       INPUTS:
3321                                    ;       OUTPUTS:
3322                                    ;       REGISTERS:      R2, R3.
3323                                    ;       CALLS:          GENPAT.
3324
3325  007452                            SETUP:: LET CMDLG :* #0                 ;CLR CMD LOGGING CODE(DISABLE LOGGING)
3326  007452  005037  003354                                                                  CLR     CMDLG
3327  007456  012137  002310                   MOV     (R1)+,CMDPKT            ;LOAD THE COMMAND WORD.
3328  007462  011137  002316                   MOV     (R1),CMDPKT+CP.CNT      ;LOAD THE BYTE/RECORD/FILE COUNT.
3329  007466  011137  003344                   MOV     (R1),BRFCNT             ;SAVE BRF FOR THIS COMMAND.
3330  007472  013702  002310                   MOV     CMDPKT,R2               ;GET CMD.
3331  007476  042702  177740                   BIC     #NCMD.C,R2              ;CLR ALL BUT CMD BITS.
3332  007502  010203                           MOV     R2,R3                   ;SAVE IT TWICE.
3333  007504  162703  000010                   SUB     #CMD.C3,R3              ;POSITION COMMAND?
3334  007510  001003                           BNE     2$                      ;BR IF NOT.
3335  007512  011137  002312                   MOV     (R1),CMDPKT+2           ;MOVE BPCR IN 2ND PKT WORD FOR POSITION
3336  007516  000461                           BR      3$
3337  007520                            2$:     IF CMDPKT EQ #WTM THEN          ;IF CMD IS A WRITE TAPE MARK THEN:
3338  007520  023727  002310  100011                                                          CMP     CMDPKT,#
3339  007526  001003                                                                        BNE     50062$
3340  007530                                    LET CMDLG :* #2                 ;WTM LOGGING CODE IS 2.
3341  007530  012737  000002  003354                                                        MOV     #2,CMDLG
3342  007536                                  ENDIF
3343  007536                                                                  50062$:
3344  007536  010203                           MOV     R2,R3
3345  007540  162 03  000001                   SUB     #CMD.CO,R3              ;IS IT A READ?
3346  007544  001017                           BNE     1$                      ;BR IF NOT.
3347  007546  013737  003336  002312           MOV     DATARD,CMDPKT+CP.ADL    ;IF SO, LOAD THE BUFFER ADDR.
3348  007554                                  IF #MOD.CO SETIN CMDPKT THEN    ;IF CMD IS A READ REV THEN:
3349  007554  032737  000400  002310                                                        BIT     #MOD.CO,
3350  007562  001404                                                                       BEQ     50063$
3351  007564                                    LET CMDLG :* #4                 ;LOGGING CODE IS 4.
3352  007564  012737  000004  003354                                                        MOV     #4,CMDLG
3353  007572                                  ELSE                            ;ELSE - IF CMD IS A READ FWD:
3354  007572  000403                                                                       BR      50064$
3355  007574                                                                  50063$:
3356  007574                                    LET CMDLG :* #6                 ;LOGGING CODE IS 6.
3357  007574  012737  000006  003354                                                        MOV     #6,CMDLG
3358  007602                                  ENDIF
3359  007602                                                                  50064$:
3360  007602  000427                           BR      3$                      ;CONTINUE.
3361  007604  010203                    1$:    MOV     R2,R3                   ;IS IT
3362  007606  162703  000004                   SUB     #CMD.C2,R3              ;A SET CHARACTERISTICS CMD?
3363  007612  001011                           BNE     4$                      ;BR IF NOT.
3364  007614                                    LET CMDPKT+CP.ADL :* #SCHBK     ;SET UP ADR LO FOR SET CHAR.
3365  007614  012737  002442  002312                                                        MOV     #SCHBK,C
3366  007622  012737  000010  002316           MOV     #SCHCNT,CMDPKT+CP.CNT   ;SET BUFFER EXTENT
3367  007630                                    LET SCHBK+6 :* (R1)             ;STORE CHARACTERISTIC CODE IN SCH BLOCK.
3368  007630  011137  002450                                                                MOV     (R1),SCH
3369  007634  000412                           BR      3$                      ;CONTINUE.
3370  007636  010203                    4$:    MOV     R2,R3                   ;IS IT
3371  007640  162703  000006                   SUB     #CMD.C1!CMD.C2,R3       ;A DIAGNOSTIC (DIA) CMD?
3372  007644  001006                           BNE     3$                      ;BR IF NOT.
3373  007646  012737  000020  002316           MOV     #DIACNT,CMDPKT+CP.CNT   ;LOAD BUFFER EXTENT.
```

```
3374  007654  012737  003334  002312          3$:     MOV     #DIABLK,CMDPKT+CP.ADL   ;LOAD BUFFER ADR LOW.
3375  007662  005721                          3$:     TST     (R1)+                   ;POINT TO N (NUMBER OF TIMES TO EXECUTE
3376  007664                                          LET NCNT1 := (R1)+              ;SAVE NUMBER OF OPERATIONS
3377  007664  012137  003342                                                          MOV     (R1)+,NC
3378  007670                                          LET NCNT := #0                  ;CLEAR OPERATION COUNTER.
3379  007670  005037  003340                                                          CLR     NCNT
3380  007674  012137  003374                          MOV     (R1)+,PATERN            ;SAVE PATTERN CODE FOR CURRENT CMD.
3381  007700  010203                                  MOV     R2,R3                   ;IS IT
3382  007702  162703  000005                          SUB     #CMD.CO!CMD.C2,R3       ;A WRITE?
3383  007706  001010                                  BNE     5$                      ;BR IF NOT.
3384  007710  013737  003334  002312                  MOV     DATAWT,CMDPKT+CP.ADL    ;LOAD WRITE BUFFER LO ORDER.
3385  007716  004737  010030                          JSR     PC,GENPAT               ;GO GENERATE THE WRITE PATTERN.
3386  007722                                          LET CMDLG := #2                 ;WRITE LOGGING CODE IS 2.
3387  007722  012737  000002  003354                                                  MOV     #2,CMDLG
3388  007730                                  5$:     IF #VFY.C SETIN CMDPKT THEN     ;IF DATA VERIFICATION IS REQUIRED:
3389  007730  032737  000100  002310                                                  BIT     #VFY.C,C
3390  007736  001407                                                                  BEQ     50065$
3391  007740                                            LET VFYFLG :B= #1             ;SET VERIFY FLAG.
3392  007740  112737  000001  003442                                                  MOVB    #1,VFYFL
3393  007746  042737  000100  002310            BIC     #VFY.C,CMDPKT                 ;CLEAR VERIFY BIT(NOT USED BY HARDWARE).
3394  007754                                          ELSE                            ;IF DATA VERIFICATION IS NOT REQUIRED:
3395  007754  000402                                                                  BR      50066$
3396  007756                                                                  50065$:
3397  007756                                            LET VFYFLG :B= #0             ;CLR VERIFY FLAG.
3398  007756  105037  003442                                                          CLRB    VFYFLG
3399  007762                                          ENDIF
3400  007762                                                                  50066$:
3401  007762                                          LET PCMDWD := CMDWRD            ;SAVE PREVIOUS CMD WORD.
3402  007762  013737  003346  003352                                                  MOV     CMDWRD,P
3403  007770                                          LET CMDWRD := CMDPKT            ;SAVE PRESENT CMD WORD.
3404  007770  013737  002310  003346                                                  MOV     CMDPKT,C
3405  007776                                          IFB SWBFLG NE #0 THEN           ;IF SWAP BYTES IS ENABLED:
3406  007776  105737  003444                                                          TSTB    SWBFLG
3407  010002  001403                                                                  BEQ     50067$
3408  010004                                            LET CMDPKT := CMDPKT SET.BY #SWB.C  ;SET SWAP BIT IN COMMAND.
3409  010004  052737  010000  002310                                                  BIS     #SWB.C,C
3410  010012                                          ENDIF
3411  010012                                                                  50067$:
3412  010012  042737  004000  002310          BIC     #BRF.C,CMDPKT                   ;CLR BRF BIT (INTERNAL ONLY).
3413  010020                                          LET CMDSAV := CMDPKT            ;SAVE 1ST WORD OF COMMAND PACKET.
3414  010020  013737  002310  003350                                                  MOV     CMDPKT,C
3415  010026  000207                          RTS     PC                              ;RETURN.
```

```
3416                                    ;       THIS SUBROUTINE SETS UP AND CALLS THE APPROPRIATE SUBROUTINE TO GENERAT
3417                                    ;       THE DESIRED PATTERN FOR THE WRITE AND WRITE/VERIFY COMMANDS.
3418                                    ;       INPUTS:
3419                                    ;       OUTPUTS:
3420                                    ;       REGISTERS:      R2, R3, R4.
3421                                    ;       CALLS:          PATR0 - PATR7
3422
3423  010030                            GENPAT:: LET R3 := PATERN SHIFT 1       ;SETUP PATTERN ROUTINE POINTER
3424  010030  013703  003374                                                            MOV     PATERN,R
3425  010034  006303                                                                    ASL     R3
3426  010036                                    LET R4 := BRFCNT - #1   ;SET LENGTH OF WRITE BFR
3427  010036  013704  003344                                                            MOV     BRFCNT,R
3428  010042  005204                                                                    INC     R4
3429  010044                                    LET R4 := R4 CLR.BY #1  ;ROUNDED UP TO NEXT WORD
3430  010044  042704  000001                                                            BIC     #1,R4
3431  010050                                    LET R4 := R4    #2      ;WITH FIRST WORD RESERVED
3432  010050  162704  000002                                                            SUB     #2,R4
3433  010054                                    LET R2 := DATAWT + #2   ;FOR RECORD COUNT
3434  010054  013702  003334                                                            MOV     DATAWT,R
3435  010060  062702  000002                                                            ADD     #2,R2
3436  010064  004773  010072                   JSR     PC,@PATTBL(R3)  ;GO GENERATE THE APPROPRIATE PATTERN.
3437  010070  000207                           RTS     PC              ;RETURN TO SETUP SUBROUTINE.
3438
3439                                    ;TS04 WRITE PATTERN LOOKUP TABLE.  USED TO JSR TO THE
3440                                    ;CORRECT DATA PATTERN GENERATING ROUTINE.
3441
3442  010072  010114                    PATTBL: PATR0
3443  010074  010152                            PATR1
3444  010076  010172                            PATR2
3445  010100  010202                            PATR3
3446  010102  010226                            PATR4
3447  010104  010240                            PATR5
3448  010106  010252                            PATR6
3449  010110  010272                            PATR7
3450  010112  010324                            PATR8
3451
3452
3453                                    ;INCREMENTING PATTERN.  0 - 377.
3454
3455  010114                            PATR0:: LET R3 := #400
3456  010114  012703  000400                                                            MOV     #400,R3
3457  010120                            1$:     LET R4 := R4    #2      ;DECREMENT WORD COUNT.
3458  010120  162704  000002                                                            SUB     #2,R4
3459  010124  100411                           BMI     2$              ;BR IF DONE.
3460  010126                                    LET (R2)+ := R3         ;STORE DATA WORD.
3461  010126  010322                                                                    MOV     R3,(R2)+
3462  010130                                    LET R3 := R3 + #1002    ;UPDATE PATTERN.
3463  010130  062703  001002                                                            ADD     #1002,R3
3464  010134                                    IF R3 EQ #1000 THEN     ;IF PATTERN HAS WRAPPED AROUND THEN:
3465  010134  020327  001000                                                            CMP     R3,#1000
3466  010140  001002                                                                    BNE     500$
3467  010142                                      LET R3 := #400          ;INIT THE PATTERN AGAIN.
3468  010142  012703  000400                                                            MOV     #400,R3
3469  010146                                    ENDIF
3470  010146                                                                  500$:
3471  010146  000764                           BR      1$              ;DO IT AGAIN.
```

F 1


```
3472  010150  000207                    2$:     RTS     PC              ;RETURN.
3473
3474                                      ;ALL ONE'S PATTERN.
3475
3476  010152  012703  177777            PATR1:: MOV     #-1,R3          ;ALL ONES PATTERN;.
3477  010156                            ZROPAT: LET R4 := R4 - #2       ;DECREMENT BYTE COUNT.
3478  010156  162704  000002                                                     SUB     #2,R4
3479  010162  100402                            BMI     1$              ;DONE?,BR IF YES.
3480  010164  010322                            MOV     R3,(R2)+        ;IF NOT LOAD NEXT BYTE WITH PATTERN.
3481  010166  000773                            BR      ZROPAT          ;DO IT AGAIN.
3482
3483  010170  000207                    1$:     RTS     PC              ;RETURN.
```

```
3484                                        ;ALL ZEROES PATTERN.
3485
3486  010172  005003                PATR2:: CLR     R3              ;CLR PATTERN REGISTER.
3487  010174  004737  010156                JSR     PC,ZROPAT       ;GO GENERATE IT.
3488  010200  000207                        RTS     PC              ;RETURN.
3489
3490                                        ;ONE BIT WALKING FROM R TO L IN A FIELD OF ZEROES.
3491
3492  010202  012703  000401        PATR3:: MOV     #401,R3         ;INIT PATTERN REGISTER.
3493  010206                        WLKZRO: LET R4 := R4   #2       ;DECREMENT WORD COUNT.
3494  010206  162704  000002                                                        SUB     #2,R4
3495  010212  100404                        BMI     1$              ;BR IF DONE.
3496  010214  010322                        MOV     R3,(R2)+        ;LOAD DATA.
3497  010216  006303                        ASL     R3              ;SHIFT PATTERN.
3498  010220  005503                        ADC     R3              ;ADD CARRY BACK INTO PATTERN.
3499  010222  000771                        BR      WLKZRO          ;DO IT AGAIN.
3500  010224  000207                1$:     RTS     PC              ;RETURN.
3501
3502                                        ;ZERO BIT WALKING FROM R TO L IN A FIELD OF 1 S.
3503
3504  010226  012703  177376        PATR4:: MOV     #177376,R3      ;INIT PATTERN REGISTER.
3505  010232  004737  010206                JSR     PC,WLKZRO       ;GO GENERATE ;IT.
3506  010236  000207                        RTS     PC              ;RETURN.
3507
3508                                        ;ALTERNATING ONE AND ZERO BITS WITH ALTERNATE BYTES
3509                                        ;COMPLEMENTED.
3510
3511  010240  012703  125125        PATR5:: MOV     #125125,R3      ;INIT PATTERN REGISTER.
3512  010244  004737  010156                JSR     PC,ZROPAT       ;GO GENERATE IT.
3513  010250  000207                        RTS     PC              ;RETURN.
3514
3515                                        ;ALTERNATING BYTES OF 000 AND 377.
3516
3517  010252  012703  177400        PATR6:: MOV     #177400,R3      ;INIT PATTERN REGISTER.
3518  010256                        1$:     LET R4 := R4   #2       ;DECREMENT WORD COUNT.
3519  010256  162704  000002                                                        SUB     #2,R4
3520  010262  100402                        BMI     2$              ;BR IF DONE.
3521  010264  010322                        MOV     R3,(R2)+        ;LOAD DATA.
3522  010266  000773                        BR      1$              ;DO IT AGAIN.
3523  010270  000207                2$:     RTS     PC              ;RETURN.
3524
3525                                        ;RANDOM PATTERN GENERATOR
3526
3527  010272                        PATR7:: LET R4 := R4   #2       ;DECREMENT WORD COUNT
3528  010272  162704  000002                                                        SUB     #2,R4
3529  010276  100411                        BMI     GIT             ;BR IF DONE.
3530  010300  063737  003362  003360        ADD     RANS,RANB
3531  010306  063737  003360  003362        ADD     RANB,RANS       ;GET NEW #.
3532  010314  013722  003362                MOV     RANS,(R2)+      ;SAVE #.
3533  010320  000764                        BR      PATR7           ;CONTINUE.
3534  010322  000207                GIT:    RTS     PC              ;RETURN
3535
3536                                ;       NO PATTERN GENERATION.
3537
3538  010324  000207                PATR8:: RTS     PC              ;RETURN.
```

```
3539                                    ;       THIS SUBROUTINE INITIATES TS04 COMMAND EXECUTION
3540                                    ;       AND CHECKS FOR TS04 RESPONSE.
3541                                    ;       INPUTS:
3542                                    ;       OUTPUTS:
3543                                    ;       REGISTERS:      R2, R3.
3544                                    ;       CALLS:          DROPU, MOVMSG, FIRSTU, NEXTU, WSSR.
3545
3546  010326                            EXCUTE:: LET TIME1 := # 1                ;INIT TIMEOUT COUNTER.
3547  010326  012737  177777  003364                                                    MOV     # 1,TIME
3548  010334                                    REPEAT                             ;WAIT
3549  010334                                                                                50071$:
3550  010334                                      LET TIME1 := TIME1   #1          ;UPDATE TIMEOUT COUNTER.
3551  010334  005337  003364                                                           DEC     TIME1
3552  010340                                      IF TIME1 EQ #0 THEN              ;IF TIMED OUT:
3553  010340  005737  003364                                                           TST     TIME1
3554  010344  001011                                                                   BNE     50072$
3555  010346  004737  011224                        JSR PC,MOVMSG                  ;MOVE CURRENT PACKET MSG.
3556  010352                                        ERRDF #2,NSSRM,STAERM          ;REPORT TS04 NOT READY
3557  010352  104455                                                                   TRAP    C$ERDF
3558  010354  000002                                                                   .WORD   2
3559  010356  004271                                                                   .WORD   NSSRM
3560  010360  005372                                                                   .WORD   STAERM
3561  010362  004737  015554                        JSR PC,DROPU                   ;DROP THE UNIT.
3562  010366  000522                                BR  EXCRTN                     ;RETURN.
3563  010370                                      ENDIF
3564  010370                                                                                50072$:
3565  010370                                    UNTIL #TS.SSR SETIN @TSSR(R5)      ;WAIT UNTIL DEVICE IS READY.
3566  010370  032775  000200  002462                                                   BIT     #TS.SSR,
3567  010376  001756                                                                   BEQ     50071$
3568  010400                                    IF CMDWRD EQ #SCH THEN             ;IF WE ARE DOING A SET CHAR CMD THEN:
3569  010400  023727  003346  140004                                                   CMP     CMDWRD,#
3570  010406  001022                                                                   BNE     50073$
3571  010410                                      LET R5SAVE := R5                 ;SAVE CURRENT DEVICE POINTER.
3572  010410  010537  003400                                                           MOV     R5,R5SAV
3573  010414  004737  015452                      JSR   PC,FIRSTU                  ;FIND FIRST UNIT.
3574  010420                                      WHILE DEVTBL(R5) NE #END DO
3575  010420                                                                                50074$:
3576  010420  026527  002532  177777                                                   CMP     DEVTBL(R
3577  010426  001405                                                                   BEQ     50075$
3578  010430  004737  011170                        JSR PC,WSSR                    ;WAIT FOR UNIT READY OR TIME OUT.
3579  010434  004737  015520                        JSR PC,NEXTU                   ;FIND NEXT UNIT.
3580  010440                                      ENDDO
3581  010440  000767                                                                   BR      50074$
3582  010442                                                                                50075$:
3583  010442                                      LET R5 := R5SAVE                 ;RESTORE CURRENT DEVICE POINTER.
3584  010442  013705  003400                                                           MOV     R5SAVE,R
3585  010446                                      LET SCHBK := MSGPKA(R5)          ;SET UP ADR OF MSG PKT IN SCH BLOCK.
3586  010446  016537  002502  002442                                                   MOV     MSGPKA(R
3587  010454                                    ENDIF
3588  010454                                                                                50073$:
3589  010454                                    LET R3 := MSGPKA(R5)             ;ADR OF THIS UNIT'S MSG PACKET.
3590  010454  016503  002502                                                           MOV     MSGPKA(R
3591  010460                                    LET R2 := #0                     ;CLR COUNTER.
3592  010460  005002                                                                   CLR     R2
3593  010462                                    WHILE R2 NE #MSGCNT DO           ;WHILE THERE ARE MORE LOCATIONS:
3594  010462                                                                                50076$:
```

```
3595  010462  020227  000016                                                                  CMP     R2,#MSGC
3596  010466  001405                                                                         BEQ     50077$
3597  010470                          LET (R3)+ := # 1                ;INIT THE MSG PACKET WITH ALL 1'S
3598  010470  012723  177777                                                                 MOV     #-1,(R3)
3599  010474                          LET R2 := R2 + #2               ;UPDATE COUNTER.
3600  010474  062702  000002                                                                 ADD     #2,R2
3601  010500                        ENDDO
3602  010500  000770                                                                         BR      50076$
3603  010502                                                                          50077$:
3604  010502  105737  002210                TSTB    DINT              ;ARE INTERRUPTS DISABLED.
3605  010506  001023                        BNE     1$                ;BR IF YES.
3606  010510                        IFB INTFLG(R5) GT #1 THEN         ;IF MORE THAN ONE INTERRUPT HAS OCCURED:
3607  010510  126527  003416  000001                                                         CMPB    INTFLG(R
3608  010516  003412                                                                       BLE     50100$
3609  010520                          LET TSSREG := @TSSR(R5)                 ;FREEZE THE CURRENT STATUS REG F
3610  010520  017537  002462  003402                                                       MOV     @TSSR(R5
3611  010526                          ERRDF #15,TOOMM,STAERM          ;REPORT TOO MANY INTERRUPTS.
3612  010526  104455                                                                       TRAP    C$ERDF
3613  010530  000017                                                                       .WORD   15
3614  010532  004460                                                                       .WORD   TOOMM
3615  010534  005372                                                                       .WORD   STAERM
3616  010536  004737  015554                JSR PC,DROPU              ;DROP THE UNIT
3617  010542  000434                        BR  EXCRTN                ;RETURN   UNIT HAS BEEN DROPPED.
3618  010544                        ENDIF
3619  010544                                                                          50100$:
3620  010544                        LET INTFLG(R5) := #0              ;CLR INTERRUPT FLAG FOR THIS DEV.
3621  010544  005065  003416                                                                 CLR     INTFLG(R
3622  010550  052737  000200  002310        BIS     #IE.C,CMDPKT      ;SET INT ENABLE BIT.
3623  010556                          1$:   IFB ERRREC EQ #0 THEN     ;IF NOT RETRYING
3624  010556  105737  003415                                                                 TSTB    ERRREC
3625  010562  001005                                                                       BNE     50101$
3626  010564                          LET RECCNT(R5) := RECCNT(R5) + #1
3627  010564  005265  003324                                                                 INC     RECCNT(R
3628  010570                          LET @DATAWT := RECCNT(R5)       ;THEN UPDATE REC COUNT TO WRITE IT ON TA
3629  010570  016577  003324  172536                                                         MOV     RECCNT(R
3630  010576                        ENDIF
3631  010576                                                                          50101$:
3632  010576  012775  002310  002452        MOV     #CMDPKT,@TSDB(R5)         ;LOAD TSDB WITH CMDPKT ADDRESS
3633                                                                          ;THIS INITIATES COMMAND EXECUTION.
3634  010604                        IF #TS.SSR SETIN @TSSR(R5) THEN ;IF READY DID NOT DROP THEN:
3635  010604  032775  000200  002462                                                         BIT     #TS.SSR,
3636  010612  001410                                                                       BEQ     50102$
3637  010614  004737  011224                JSR   PC,MOVMSG                  ;MOVE CURRENT MESSAGE PACKET TO COMMON
3638  010620                                ERRDF #3,TOERM,STAERM            ;REPORT NO TS04 RESPONSE.
3639  010620  104455                                                                       TRAP    C$ERDF
3640  010622  000003                                                                       .WORD   3
3641  010624  004207                                                                       .WORD   TOERM
3642  010626  005372                                                                       .WORD   STAERM
3643  010630  004737  015554                JSR PC,DROPU                     ;DROP THE UNIT
3644  010634                        ENDIF
3645  010634                                                                          50102$:
3646  010634  000207          EXCRTN: RTS     PC                        ;RETURN.
```

```
3647                            ;       THIS SUBROUTINE WAITS FOR THE TS04 INERRUPT OR DONE BIT TO SET AND ALLOW
3648                            ;       OPERATOR TO TRANSFER CONROL TO THE SUPERVISOR.
3649                            ;       UPON APPEARANCE OF THE INTERRUPT OR DONE, CHECK TSSR FOR STATUS ERRORS,
3650                            ;       LOG BYTES AND ERRORS AND PERFORM ERROR RECOVERY IF NESSASARY.
3651                            ;       INPUTS:
3652                            ;       OUTPUTS:
3653                            ;       REGISTERS:      R2, R3.
3654                            ;       CALLS:          DROPU, MOVMSG, RECUD, CHKERR, LOG, CLRERR.
3655
3656  010636                    GOWAIT::IF DEVTBL'R5) EQ #NINUSE THEN   ;BTL
3657  010636  026527  002532  177774                                                  CMP     DEVTBL'R
3658  010644  001003                                                                  BNE     50103$
3659  010646  000137  011152                   JMP 1$                                ;BTL
3660  010652                                    ELSE                                  ;BTL
3661  010652  000400                                                                  BR      50104$
3662  010654                                                                  50103$:
3663  010654                                    ENDIF                                 ;BTL
3664  010654                                                                  50104$:
3665  010654                                    LET TIME1 := # 1                      ;INIT TIME OUT COUNTER.
3666  010654  012737  177777  003364                                                  MOV     # 1,TIME
3667  010662                                    REPEAT                                ;REPEAT UNTIL INTERRUPT OCCURES:
3668  010662                                                                  50105$:
3669  010662                                      BREAK                               ;GO TO THE SUPER TO ALLOW TTY INPUT.
3670  010662  104422                                                                  TRAP    C$BRK
3671  010664                                      IF CMDWRD EQ #RWD THEN              ;IF COMMAND WAS REWIND THEN:
3672  010664  023727  003346  102010                                                  CMP     CMDWRD,#
3673  010672  001014                                                                  BNE     50106$
3674  010674                                        DELAY 10.                         ;WAIT  EXTRA MSECS EACH LOOP.
3675  010674  012727  000012                                                          MOV     #10.,(PC
3676  010700  000000                                                                  .WORD   0
3677  010702  013727  002116                                                          MOV     L$DLY,(P
3678  010706  000000                                                                  .WORD   0
3679  010710  005367  177772                                                          DEC      6(PC)
3680  010714  001375                                                                  BNE     .-4
3681  010716  005367  177756                                                          DEC     -22(PC)
3682  010722  001367                                                                  BNE     .-20
3683  010724                                      ENDIF
3684  010724                                                                  50106$:
3685  010724                                      IF CMDWRD EQ #SFF OR CMDWRD EQ #SFR THEN
3686  010724  023727  003346  105010                                                  CMP     CMDWRD,#
3687  010732  001404                                                                  BEQ     50107$
3688  010734  023727  003346  105410                                                  CMP     CMDWRD,#
3689  010742  001014                                                                  BNE     50110$
3690  010744                                                                  50107$:
3691  010744                                        DELAY 12.                         ;ADD DELAY FOR SPACE TAPE MARK COMMANDS
3692  010744  012727  000014                                                          MOV     #12.,(PC
3693  010750  000000                                                                  .WORD   0
3694  010752  013727  002116                                                          MOV     L$DLY,(P
3695  010756  000000                                                                  .WORD   0
3696  010760  005367  177772                                                          DEC      6(PC)
3697  010764  001375                                                                  BNE     .-4
3698  010766  005367  177756                                                          DEC     -22(PC)
3699  010772  001367                                                                  BNE     .-20
3700  010774                                      ENDIF
3701  010774                                                                  50110$:
3702  010774                                      IFB DINT EQ #0 THEN                 ;IF INTERRUPTS ARE ENABLED.
```

```
3703  010774  105737  002210                                                    TSTB    DINT
3704  011000  001003                                                          BNE     50111$
3705  011002                    LET R2 := INTFLG(R5)        ;FETCH INTERRUPT OCCURRED FLAG.
3706  011002  016502  003416                                                  MOV     INTFLG(R
3707  011006                  ELSE                          ;IF IN BRUTUS MODE:
3708  011006  000406                                                          BR      50112$
3709  011010                                                              50111$:
3710  011010                    LET R3 := COMP #TS.SSR      ;SET UP A MASK FOR THE DONE BIT.
3711  011010  012703  000200                                                  MOV     #TS.SSR,
3712  011014  005103                                                          COM     R3
3713  011016                    LET R2 := @TSSR(R5) CLR.BY R3  ;FETCH DONE BIT.
3714  011016  017502  002462                                                  MOV     @TSSR(R5
3715  011022  040302                                                          BIC     R3,R2
3716  011024                  ENDIF
3717  011024                                                              50112$:
3718  011024                    LET TIME1 := TIME1 - #1     ;UPDATE TIMEOUT COUNTER.
3719  011024  005337  003364                                                  DEC     TIME1
3720  011030                UNTIL R2 NE #0 OR TIME1 EQ #0   ;REPEAT UNTIL INTERRUPT OR READY OCCURES
3721  011030  005702                                                          TST     R2
3722  011032  001003                                                          BNE     50113$
3723  011034  005737  003364                                                  TST     TIME1
3724  011040  001310                                                          BNE     50105$
3725  011042                                                              50113$:
3726  011042                  IF TIME1 EQ #0 THEN           ;IF TIME OUT HAS OCCURRED:
3727  011042  005737  003364                                                  TST     TIME1
3728  011046  001022                                                          BNE     50114$
3729  011050                    LET @DATAWT := RECCNT(R5)   #1 ;RE ADJUST REC COUNT DOWN
3730  011050  016577  003324  172256                                          MOV     RECCNT(R
3731  011056  005377  172252                                                  DEC     @DATAWT
3732  011062  004737  011224    JSR   PC,MOVMSG             ;MOVE CURRENT MSG PACKET TO COMMON AREA.
3733  011066                    ERRDF #4,NOINTM,STAERM      ;REPORT NO INTERRUPT.
3734  011066  104455                                                          TRAP    C$ERDF
3735  011070  000004                                                          .WORD   4
3736  011072  004421                                                          .WORD   NOINTM
3737  011074  005372                                                          .WORD   STAERM
3738  011076  004737  015554    JSR PC,DROPU                ;DROP THE UNIT.
3739  011102                    LET R3 := #ENDERF
3740  011102  012703  003416                                                  MOV     #ENDERF,
3741  011106  004737  011154    JSR PC,CLRERR                   ;CLEAR ALL ERROR FLAGS
```

```
3742
3743  011112                                        ELSE
3744  011112  000417                                                        BR      50115$
3745  011114                                                          50114$:
3746  011114  004737  011224                          JSR   PC,MOVMSG     ;MOVE CURRENT MSG. PACKET TO COMMON AREA
3747  011120  004737  011310                          JSR   PC,RECUD      ;UPDATE THE RECORD COUNT.
3748  011124  004737  011456                          JSR   PC,CHKERR     ;CHECK FOR STATUS ERRORS.
3749  011130                                          IFB WRTYFG EQ #0 THEN ;
3750  011130  105737  003407                                                TSTB    WRTYFG
3751  011134  001006                                                         BNE     50116$
3752  011136  004737  014102                            JSR PC,LOG        ;LOG BYTES AND ERRORS.
3753  011142                                            LET R3 := #ENDERF
3754  011142  012703  003416                                                MOV     #ENDERF,
3755  011146  004737  011154                            JSR PC,CLRERR     ;CLEAR ALL ERROR FLAGS
3756  011152                                          ENDIF
3757  011152                                                          50116$:
3758  011152                                        ENDIF
3759  011152                                                          50115$:
3760  011152  000207                         1$:     RTS     PC        ;RETURN IF DONE.
```

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

011154                  CLRERR:: LET R2 := #BGNFLG
011154  012702  003404                                          MOV     #BGNFLG,
011160                          REPEAT
011160                                                  50117$:
011160                            LET (R2)+ :  #0
011160  005022                                          CLR     (R2)+
011162                          UNTIL R2 EQ R3
011162  020203                                          CMP     R2,R3
011164  001375                                          BNE     50117$
011166  000207                  RTS PC


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

011170                  WSSR::  LET TIME1 := # 1                ;INIT TIMEOUT COUNTER.
011170  012737  177777  003364                                  MOV     #-1,TIME
011176                          REPEAT                          ;REPEAT UNTIL DEV READY OR TIMEOUT:
011176                                                  50120$:
011176                            BREAK                           ;BREAK TO THE SUPERVISOR.
011176  104422                                          TRAP    C$BRK
011200                            LET TIME1 := TIME1    #1        ;UPDATE TIMEOUT COUNTER.
011200  005337  003364                                          DEC     TIME1
011204                          UNTIL #TS.SSR SETIN @TSSR(R5) OR TIME1 EQ #0
011204  032775  000200  002462                                  BIT     #TS.SSR,
011212  001003                                                  BNE     50121$
011214  005737  003364                                          TST     TIME1
011220  001366                                                  BNE     50120$
011222                                                  50121$:
                                                        ;REPEAT UNTIL DEV READY OR TIMEOUT.
011222  000207                  RTS PC                  ;RETURN.
```

```
                                ;       SUBROUTINE TO MOVE THE CURRENT MESSAGE PACKET TO THE COMMON AREA AND
                                ;       TO UPDATE THE CURRENT TERMINATION CLASS CODE.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R3.
                                ;       CALLS:

011224                          MOVMSG:: LET TSSREG := @TSSR(R5)        ;FREEZE THE STATUS REG CONTENTS
011224  017537  002462  003402                                                   MOV     @TSSR(R5
011232                                  LET R2 := TSSREG CLR.BY #TSC.TCC ;EXTRACT THE TERMINATION CLASS CODE.
011232  013702  003402                                                          MOV     TSSREG,R
011236  042702  177761                                                          BIC     #TSC.TCC
011242                                  LET CTCC := R2 SHIFT -1         ;AND SAVE IT
011242  010237  003376                                                          MOV     R2,CTCC
011246  006237  003376                                                          ASR     CTCC
011252                                  LET R3 := MSGPKA(R5)            ;ADR OF THIS DEVICE'S MSG.
011252  016503  002502                                                          MOV     MSGPKA(R
011256                                  LET R2 := #0                    ;CLR COUNTER.
011256  005002                                                          CLR     R2
011260                                  WHILE R2 NE #MSGCNT DO          ;WHILE THERE ARE MORE LOCATIONS:
011260                                                                  50122$:
011260  020227  000016                                                          CMP     R2,#MSGC
011264  001405                                                          BEQ     50123$
011266                                    LET MSGPKT(R2) := (R3)+       ;MOVE MSG TO COMMON AREA.
011266  012362  002334                                                          MOV     (R3)+,MS
011272                                    LET R2 := R2 + #2             ;UPDATE COUNTER.
011272  062702  000002                                                          ADD     #2,R2
011276                                  ENDDO
011276  000770                                                          BR      50122$
011300                                                                  50123$:
011300                          LET EOTFLG(R5) := MSGPKT+MS.XSO ;MOVE XSTATO TO EOT FLAG.
011300  013765  002342  003426                                                   MOV     MSGPKT+M
011306  000207                  RTS     PC
```

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

011310                                  RECUD:: IFB RECLOG EQ #0 THEN           ;IF RECORD HAS NOT BEEN LOGGED:
011310  105737  003411                                                                  TSTB    RECLOG
011314  001057                                                                        BNE     50124$
011316                                    LET RECCNT(R5) := RECCNT(R5)  #1
011316  005365  003324                                                                DEC     RECCNT(R
011322                                    IF #BIT0 NOTSETIN CTCC AND #X2.OPM SETIN MSGPKT+MS.XS2 THEN   ;IF TAPE
011322  032737  000001  003376                                                        BIT     #BIT0,C
011330  001046                                                                        BNE     50125$
011332  032737  100000  002346                                                        BIT     #X2.OPM,
011340  001442                                                                        BEQ     50125$
011342                                      LET RECLOG :B= RECLOG + #1  ;SET RECORD LOGGED.
011342  105237  003411                                                                INCB    RECLOG
011346                                      IF CMDWRD EQ #RWD THEN      ;IF THIS IS A REWIND CMD:
011346  023727  003346  102010                                                        CMP     CMDWRD,#
011354  001003                                                                        BNE     50126$
011356                                        LET RECCNT(R5) := #0       ;CLEAR RECORD COUNT.
011356  005065  003324                                                                CLR     RECCNT(R
011362                                      ELSE
011362  000431                                                                        BR      50127$
011364                                                                          50126$:
011364                                        IF #BRF.C SETIN CMDWRD THEN       ;IF BRF USED, UPDATE RECORD COUN
011364  032737  004000  003346                                                        BIT     #BRF.C,C
011372  001425                                                                        BEQ     50130$
011374                                          IF #MOD.CO NOTSETIN CMDWRD THEN  ;IF A FORWARD CMD:
011374  032737  000400  003346                                                        BIT     #MOD.CO,
011402  001007                                                                        BNE     50131$
011404                                    IF #MOD.CO NOTSETIN PCMDWD THEN  ;IF PREV CMD WAS A FWD ALSO:
011404  032737  000400  003352                                                        BIT     #MOD.CO,
011412  001002                                                                        BNE     50132$
011414                                      LET RECCNT(R5) := RECCNT(R5) + #1  ;INCREMENT RECORD COUNT.
011414  005265  003324                                                                INC     RECCNT(R
011420                                    ENDIF
011420                                                                          50132$:
011420                                  ELSE                        ;IF REVERSE CMD:
011420  000412                                                                        BR      50133$
011422                                                                          50131$:
011422                                    IF #MOD.CO SETIN PCMDWD THEN  ;IF PREVIOUS CMD WAS A REV ALSO:
011422  032737  000400  003352                                                        BIT     #MOD.CO,
011430  001406                                                                        BEQ     50134$
011432                                      IF #X0.BOT NOTSETIN EOTFLG(R5) THEN ;WHEN NOT AT BOT THEN
011432  032765  000002  003426                                                        BIT     #X0.BOT,
011440  001002                                                                        BNE     50135$
011442                                        LET RECCNT(R5) := RECCNT(R5)  #1  ;DECREMENT RECORD COUNT
011442  005365  003324                                                                DEC     RECCNT(R
011446                                      ENDIF
011446                                                                          50135$:
011446                                    ENDIF
011446                                                                          50134$:
011446                                  ENDIF
011446                                                                          50133$:
```

```
3894  011446                                        ENDIF
3895  011446                                                                501305:
3896  011446                                      ENDIF
3897  011446                                                                501275:
3898  011446                                    ENDIF
3899  011446                                                                501255:
3900  011446                                    LET @DATAWT := RECCNT(R5)
3901  011446  016577  003324  171660                                              MOV     RECCNT(R
3902  011454                                  ENDIF
3903  011454                                                                501245:
3904  011454  000207                          RTS     PC                   ;RETURN.
```

```
3905                                    ;       THIS IS THE ERROR CHECK SUBROUTINE.  AFTER INTERRUPT THIS
3906                                    ;       SUBROUTINE IS CALLED TO CHECK THE TS04 STATUS.
3907                                    ;       IF SPECIAL COND IS SET THEN THE TCC HANDLING SUBROUTINE IS ENTERED.
3908                                    ;       IF THE RFC IS NON ZERO FOR A COMMAND REQUIRING A BPCR,
3909                                    ;       THEN AN ERROR RFC IS REPORTED.
3910                                    ;       INPUTS:
3911                                    ;       OUTPUTS:
3912                                    ;       REGISTERS:      R2, R4.
3913                                    ;       CALLS:          TCC0 TCC7.
3914
3915  011456                            CHKERR:: IF DEVTBL(R5) EQ #NINUSE THEN  ;BTL
3916  011456  026527  002532  177774                                                    CMP     DEVTBL(R
3917  011464  001003                                                                   BNE     50136$
3918  011466  000137  011676                   JMP 1$                          ;BTL
3919  011472                                   ELSE                            ;BTL
3920  011472  000400                                                                   BR      50137$
3921  011474                                                                  50136$:
3922  011474                                   ENDIF                           ;BTL
3923  011474                                                                  50137$:
3924  011474                                   IF #TS.SC SETIN TSSREG THEN             ;IF SPECIAL COND STATUS IS SET T
3925  011474  032737  100000  003402                                                    BIT     #TS.SC,T
3926  011502  001441                                                                   BEQ     50140$
3927  011504                                     IF CTCC NE #2 THEN              ;IF TCC IS NOT 2 THEN:
3928  011504  023727  003376  000002                                                    CMP     CTCC,#2
3929  011512  001405                                                                   BEQ     50141$
3930  011514                                       IFB ERRREC EQ #0 THEN         ;IF NOT IN ERROR RECOVERY:
3931  011514  105737  003415                                                          TSTB    ERRREC
3932  011520  001002                                                                   BNE     50142$
3933  011522  005265  003264                        INC     SCCNT(R5)           ;INC SC COUNTER.
3934  011526                                       ENDIF
3935  011526                                                                  50142$:
3936  011526                                     ENDIF
3937  011526                                                                  50141$:
3938  011526                                   IF #TS.NXM SETIN TSSREG OR #TS.UPE SETIN TSSREG THEN  ;WHEN NON-EXISTA
3939  011526  032737  004000  003402                                                    BIT     #TS.NXM,
3940  011534  001004                                                                   BNE     50143$
3941  011536  032737  040000  003402                                                    BIT     #TS.UPE,
3942  011544  001412                                                                   BEQ     50144$
3943  011546                                                                  50143$:
3944  011546                                     IF #X2.OPM NOTSETIN MSGPKT.MS.XS2 THEN        ;AND TAPE NOT MO
3945  011546  032737  100000  002346                                                    BIT     #X2.OPM,
3946  011554  001003                                                                   BNE     50145$
3947  011556                                       LET R2 := #5                  ;SET TCC5 INDEX
3948  011556  012702  000005                                                          MOV     #5,R2
3949  011562                                     ELSE
3950  011562  000402                                                                   BR      50146$
3951  011564                                                                  50145$:
3952  011564                                       LET R2 := #4                  ;TAPE MOVED, SET TCC4 INDEX
3953  011564  012702  000004                                                          MOV     #4,R2
3954  011570                                     ENDIF
3955  011570                                                                  50146$:
3956  011570                                   ELSE
3957  011570  000402                                                                   BR      50147$
3958  011572                                                                  50144$:
3959  011572                                     LET R2 := CTCC                ;SET DETECTED TCC INDEX
3960  011572  013702  003376                                                          MOV     CTCC,R2
```

```
3961  011576                                                ENDIF
3962  011576                                                                                50147$:
3963  011576                                                LET R2 := R2 SHIFT 1  ;CURRENT TCC X 2.
3964  011576  006302                                                                       ASL     R2
3965  011600  004772  011700                               JSR   PC,@TCCRA(R2)           ;GO TO THE TCC HANDLING SUBROUTINE.
3966  011604                                              ELSE
3967  011604  000426                                                                       BR      50150$
3968  011606                                                                                50140$:
3969  011606                                                IF @BRF.C SETIN CMDWRD THEN   ;IF BRF IS USED IN THIS CMD THEN:
3970  011606  032737  004000  003346                                                       BIT     @BRF.C,C
3971  011614  001422                                                                       BEQ     50151$
3972  011616                                                  IF MSGPKT.MS.RFC NE @0 THEN   ;IF THERE IS AN RFC THEN:
3973  011616  005737  002340                                                             TST     MSGPKT.M
3974  011622  001417                                                                       BEQ     50152$
3975  011624                                                    IFB RANDOM EQ @0 ORB VFYFLG NE @0 THEN
3976  011624  105737  003441                                                             TSTB    RANDOM
3977  011630  001403                                                                     BEQ     50153$
3978  011632  105737  003442                                                             TSTB    VFYFLG
3979  011636  001411                                                                     BEQ     50154$
3980  011640                                                                                50153$:
3981                                                                                    ;IF NOT IN RANDOM OR IF CMD IS WTV:
3982  011640                                                      IFB IRE EQ @0 THEN          ;IF RFC ERROR REPORTS ARE ALLOWED:
3983  011640  105737  003445                                                             TSTB    IRE
3984  011644  001006                                                                     BNE     50155$
3985  011646                                                        LET HRDCNT(R5) := HRDCNT(R5) + @1  ;UPDATE HARD ERROR COUNT
3986  011646  005265  003304                                                             INC     HRDCNT(R
3987  011652                                                        ERRHRD @13,RFCERM,STAERM      ;REPORT RFC ERROR
3988  011652  104456                                                                     TRAP    C$ERHRD
3989  011654  000015                                                                     .WORD   13
3990  011656  004254                                                                     .WORD   RFCERM
3991  011660  005372                                                                     .WORD   STAERM
3992  011662                                                        ENDIF
3993  011662                                                                                50155$:
3994  011662                                                      ENDIF
3995  011662                                                                                50154$:
3996  011662                                                    ENDIF
3997  011662                                                                                50152$:
3998  011662                                                  ENDIF
3999  011662                                                                                50151$:
4000  011662                                                ENDIF
4001  011662                                                                                50150$:
4002  011662                                                IFB RWERR NE @0 THEN          ;IF A READ/WRITE ERROR HAS OCCURRED THEN
4003  011662  105737  003413                                                             TSTB    RWERR
4004  011666  001403                                                                     BEQ     50156$
4005  011670                                                  LET CMDPKT := CMDSAV          ;RESTORE CMD PACKET AFTER ERROR RECOV.
4006  011670  013737  003350  002310                                                     MOV     CMDSAV,C
4007  011676                                                ENDIF
4008  011676                                                                                50156$:
4009  011676  000207                                1$:    RTS     PC                    ;RETURN.
```

```
4010                                    ;       ADDRESSES OF TCC HANDLING ROUTINES FOR TERMINATION CLASS CODES 0 - 7.
4011
4012  011700  011720                    TCCRA:  TCC0
4013  011702  011736                            TCC1
4014  011704  011754                            TCC2
4015  011706  012064                            TCC3
4016  011710  012102                            TCC4
4017  011712  012552                            TCC5
4018  011714  012650                            TCC6
4019  011716  012700                            TCC7
```

```
                                        ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 0, UNDEFINED SPECIAL
                                        ;       CONDITION ERROR.
                                        ;       INPUTS:
                                        ;       OUTPUTS:
                                        ;       REGISTERS:
                                        ;       CALLS:
                                        TCC0::  LET HRDCNT(R5) := HRDCNT(R5) + #1  ;UPDATE HARD ERROR COUNT.
011720  005265  003304                                                              INC     HRDCNT(R
                                                ERRHRD #5,SCERM,STAERM          ;REPORT SPECIAL CONDITION ERROR.
011724                                                                          TRAP    C$ERHRD
011724  104456                                                                    .WORD   5
011726  000005                                                                  .WORD   SCERM
011730  004230                                                                  .WORD   STAERM
011732  005372
011734  000207                                  RTS PC                          ;RETURN.
                                        ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 1, ATTENTION CONDITION.
                                        ;       THIS TCC INDICATES THAT THE DRIVE HAS UNDERGONE A STATUS CHANGE
                                        ;       SUCH AS GOING OFFLINE OR COMING ONLINE.
                                        ;       INPUTS:
                                        ;       OUTPUTS:
                                        ;       REGISTERS:      R2,R4
                                        ;       CALLS:          DROPU
                                        TCC1::  ERRDF #6,ATTNM,STAERM           ;REPORT ATTENTION-UNIT OFF LINE.
011736                                                                          TRAP    C$ERDF
011736  104455                                                                    .WORD   6
011740  000006                                                                  .WORD   ATTNM
011742  004335                                                                  .WORD   STAERM
011744  005372
011746  004737  015554                          JSR PC,DROPU                    ;DROP THE UNIT.
011752  000207                                  RTS PC                          ;RETURN.
```

```
4055                                    ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 2, TAPE STATUS ALERT.
4056                                    ;       A STATUS CONDITION HAS BEEN ENCOUNTERED THAT MAY HAVE SIGNIFICANCE
4057                                    ;       TO THE PROGRAM.  BITS OF INTEREST INCLUDE TMK, RLS, LET, RLL, BOT, EOT.
4058                                    ;       INPUTS:
4059                                    ;       OUTPUTS:
4060                                    ;       REGISTERS:
4061                                    ;       CALLS:
4062
4063  011754                            TCC2::    IF @XO.BOT SETIN MSGPKT+MS.XSO ANDB EXPBOT NE @0 THEN
4064  011754  032737  000002  002342                                                          BIT     @XO.BOT,
4065  011762  001404                                                                        BEQ     50157$
4066  011764  105737  003440                                                              TSTB    EXPBOT
4067  011770  001401                                                                        BEQ     50157$
4068                                                                          ;IF AT BOT AND BOT IS EXPECTED:
4069  011772  000433                          BR TC2RTN                       ;RETURN TCC2 CAUSED BY EXPECTED BOT.
4070  011774                                ENDIF
4071  011774                                                                          50157$:
4072  011774                                IF @XO.RLS!XO.RLL!XO.TMK!XO.LET!XO.BOT SETIN MSGPKT+MS.XSO THEN
4073  011774  032737  170002  002342                                                          BIT     @XO.RLS!
4074  012002  001427                                                                        BEQ     50160$
4075                                                                          ;IF TCC2 CAUSED BY ANYTHING BUT EOT:
4076  012004                              IFB RANDOM EQ @0 ORB VFYFLG NE @0 THEN
4077  012004  105737  003441                                                              TSTB    RANDOM
4078  012010  001403                                                                        BEQ     50161$
4079  012012  105737  003442                                                              TSTB    VFYFLG
4080  012016  001421                                                                        BEQ     50162$
4081  012020                                                                          50161$:
4082                                                                          ;IF NOT IN RANDOM OR IF CMD IS WTV:
4083  012020                                IFB IRE EQ @0 THEN                        ;IF RFC ERROR REPORTS ARE ALLOWED:
4084  012020  105737  003445                                                              TSTB    IRE
4085  012024  001016                                                                        BNE     50163$
4086  012026                                IFB ERRREC NE @0 THEN                     ;IF WE ARE IN ERROR RECOVERY THE
4087  012026  105737  003415                                                                  TSTB    ERRREC
4088  012032  001403                                                                        BEQ     50164$
4089  012034                                  LET UNREC :B  UNREC + #1                  ;SET UNRECOVERABLE FLAG FOR LO
4090  012034  105237  003414                                                                  INCB    UNREC
4091  012040                                ELSE                                      ;ELSE - IF NOT IN ERROR RECOVERY
4092  012040  000402                                                                        BR      50165$
4093  012042                                                                          50164$:
4094  012042                                  LET SCCNT(R5) := SCCNT(R5) + #1 ;INCREMENT THE SPEC COND COUNT
4095  012042  005265  003264                                                                INC     SCCNT(R5
4096  012046                                ENDIF
4097  012046                                                                          50165$:
4098  012046                                  LET HRDCNT(R5) := HRDCNT(R5) + #1   ;UPDATE HARD ERROR COUNT.
4099  012046  005265  003304                                                                INC     HRDCNT(R
4100  012052                                  ERRHRD #7,TSAM,STAERM             ;REPORT TAPE STATUS ALERT.
4101  012052  104456                                                                        TRAP    C$ERHRD
4102  012054  000007                                                                        .WORD   7
4103  012056  004436                                                                        .WORD   TSAM
4104  012060  005372                                                                        .WORD   STAERM
4105  012062                              ENDIF
4106  012062                                                                          50163$:
4107  012062                            ENDIF
4108  012062                                                                          50162$:
4109  012062                          ENDIF
4110  012062                                                                          50160$:
```

```
GLOBAL AREAS    MACY11 30(1046)  06 APR 84  08:51  PAGE 101
CZTSHD.P11    06-APR-84 08:49          GLOBAL SUBROUTINES SECTION                                    SEQ 0099

 4111  012062  000207                        TC2RTN:   RTS PC                              ;RETURN.
 4112
 4113
 4114
 4115
 4116
 4117                                        ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 3, FUNCTION REJECT.
 4118                                        ;       THE SPECIFIED FUNCTION WAS NOT INITIATED.  BITS OF INTEREST ARE
 4119                                        ;       RMR, OFL, VCK, BOT, ILC, WLE, ILA, AND NBA.
 4120                                        ;       INPUTS:
 4121                                        ;       OUTPUTS:
 4122                                        ;       REGISTERS:      R2,R4
 4123                                        ;       CALLS:          DROPU
 4124
 4125  012064                                TCC3::  ERRDF #8,FUNRM,STAERM         ;REPORT FUNCTION REJECT.
 4126  012064  104455                                                                      TRAP    C$ERDF
 4127  012066  000010                                                                      .WORD   8
 4128  012070  004353                                                                      .WORD   FUNRM
 4129  012072  005372                                                                      .WORD   STAERM
 4130  012074  004737  015554                       JSR PC,DROPU                  ;DROP THE UNIT.
 4131  012100  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. ON REACHING 20 BAD
                                        ;       SPOTS LOGGED, A BAD TAPE OVERFLOW MSG IS PRINTED AND THE
                                        ;       UNIT DROPPED.
                                        ;       THE SECOND ALGORITHM ISSUES THE TS11 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

012102                                  TCC4::  IF DEVTBL(R5) EQ #NINUSE THEN   ;BTL
012102  026527  002532  177774                                                           CMP     DEVTBL(R
012110  001003                                                                           BNE     50166$
012112  000137  012550                          JMP 3$                                  ;BTL
012116                                          ELSE                                    ;BTL
012116  000400                                                                           BR      50167$
012120                                                                          50166$:
012120                                          ENDIF                                   ;BTL
012120                                                                          50167$:
012120                                          IF CMDLG EQ #2 ANDB BADTSW NE #0 THEN
012120  023727  003354  000002                                                           CMP     CMDLG,#2
012126  001134                                                                           BNE     50170$
012130  105737  002206                                                                   TSTB    BADTSW
012134  001531                                                                           BEQ     50170$
012136                                  IFB ERRREC EQ #0 ANDB ERCVER NE #0 THEN
012136  105737  003415                                                                   TSTB    ERRREC
012142  001007                                                                           BNE     50171$
012144  105737  002205                                                                   TSTB    ERCVER
012150  001404                                                                           BEQ     50171$
012152                                    ERRSOFT #9,RERM,STAERM        ;
012152  104457                                                                           TRAP    C$ERSOFT
012154  000011                                                                           .WORD   9
012156  004550                                                                           .WORD   RERM
012160  005372                                                                           .WORD   STAERM
012162                                  ENDIF
012162                                                                          50171$:
012162                                  IFB IREC EQ #0 THEN             ;
012162  105737  002211                                                                   TSTB    IREC
012166  001111                                                                           BNE     50172$
012170                                    LET ERRREC :B= ERRREC + #1     ;RETRY FLAG FOR EXCUTE SUBR: DON'T UPDAT
012170  105237  003415                                                                   INCB    ERRREC
012174                                    LET WRTYER :B= WRTYER + #1     ;REWRITE ERROR FLAG FOR WRTY SUBR
012174  105237  003410                                                                   INCB    WRTYER
012200                                    IFB WRTYFG EQ #0 THEN          ;FIRST RETRY ON THIS RECORD: SUBSEQUENT
012200  105737  003407                                                                   TSTB    WRTYFG
012204  001101                                                                           BNE     50173$
                                                                        ;RETRIES WITH TCC4 ERRORS BY PASS THIS S
012206                                      LET WTYWRD :- CMDWRD        ;SAVE WRITE COMMAND PACKET
```

```
4188  012206  013737  003346  013366                                                    MOV     CMDWRD,W
4189  012214
4190  012214  013737  002310  013364                                                    MOV     CMDPKT,W
4191  012222
4192  012222  013737  002316  013370                                                    MOV     CMDPKT+C
4193  012230
4194  012230  105237  003413                                                          INCB    RWERR
4195  012234
4196  012234  105237  003407                                                          INCB    WRTYFG
4197  012240
4198  012240                                                                        50174$:
4199  012240                                                                        ;COUNT GLOBAL WRITE RETR
4200  012240  005265  003244                                                          INC     WRTYCT(R
4201  012244
4202  012244  005037  003404                                                          CLR     RETRYC
4203  012250
4204  012250  105037  003406                                                          CLRB    RPTCNT
4205  012254  004737  013044
4206  012260
4207  012260  026527  002532  177774                                                  CMP     DEVTBL(R
4208  012266  001003                                                                  BNE     50175$
4209  012270  000137  012550
4210  012274
4211  012274  000400                                                                  BR      50176$
4212  012276
4213  012276                                                                        50175$:
4214  012276
4215  012276                                                                        50176$:
4216  012276  105737  003410                                                          TSTB    WRTYER
4217  012302  001404                                                                  BEQ     50177$
4218  012304  027727  171126  000050                                                  CMP     @BTPT,#4
4219  012312  103752                                                                  BLO     50174$
4220  012314
4221                                                                                50177$:
4222  012314
4223  012314  027727  171116  000050                                                  CMP     @BTPT,#4
4224  012322  103423                                                                  BLO     50200$
4225  012324
4226  012324  012746  013457                                                          MOV     #BTMSG2,
4227  012330  012746  000001                                                          MOV     #1,(SP)
4228  012334  010600                                                                  MOV     SP,R0
4229  012336  104414                                                                  TRAP    C$PNTB
4230  012340  062706  000004                                                          ADD     #4,SP
4231  012344  004737  013576
4232  012350
4233  012350  005365  003324                                                          DEC     RECCNT(R
4234  012354  004737  015554
4235  012360
4236  012360  005065  003324                                                          CLR     RECCNT(R
4237  012364
4238  012364  012775  002330  002452                                                  MOV     #RWCPK,@
4239  012372
4240  012372                                                                        50200$:
4241  012372
4242  012372  105037  003407                                                          CLRB    WRTYFG
4243  012376
```

```
LET WTYCMD := CMDPKT      ;

LET WTYBRF := CMDPKT+CP.CNT       ;

LET RWERR :B= RWERR + #1  ;LOG SUBR FLAG: COUNT WRT ERRORS

LET WRTYFG :B= WRTYFG + #1        ;RETRY IN PROGRESS FLAG

REPEAT

  LET WRTYCT(R5) := WRTYCT(R5) + #1       ;COUNT GLOBAL WRITE RETR

  LET RETRYC := #0                ;CLEAR # OF RETRIES PER RECORD

  LET RPTCNT :B= #0               ;CLEAR # OF REPEATS

  JSR PC,WRTY             ;CALL WRITE RETRY
  IF DEVTBL(R5) EQ #NINUSE THEN   ;BTL

  JMP 3$                          ;BTL
  ELSE                            ;BTL

  ENDIF                           ;BTL

UNTILB WRTYER EQ #0 OR @BTPT HIS #40.     ;REPEAT RETRIES ON SAME

                                  ;UNTIL RECOVERED OR 20 BAD SPOTS
IF @BTPT HIS #40. THEN    ;WHEN 20 BAD SPOTS LOGGED

  PRINTB #BTMSG2          ;PRINT BAD TAPE OVERFLOW MSG

  JSR PC,BORERS           ;ERASE BAD RECORD
  LET RECCNT(R5) := RECCNT(R5)   #1       ;

  JSR PC,DROPU            ;DROP UNIT
  LET RECCNT(R5) :=  #0   ;

  LET @TSDB(R5) := #RWCPK ;REWIND UNIT

ENDIF

LET WRTYFG :B= #0         ;RETRY COMPLETE FLAG

LET MISCFG :B  MISCFG + #1        ;DO NOT HALT ON THIS CMD FLG
```

```
4244  012376  105237  003455                                                    INCB    MISCFG
4245  012402                    LET PCMDWD := WTYWRD      ;RESTORE ORIGINAL WRT CMD AFTER RECOVERY
4246  012402  013737  013366  003352                                          MOV     WTYWRD,P
4247  012410                    ENDIF
4248  012410                                                              50173$:
4249  012410                  ELSE
4250  012410  000402                                                          BR      50201$
4251  012412                                                              50172$:
4252  012412                    LET UNREC :B= UNREC + #1    ;
4253  012412  105237  003414                                                  INCB    UNREC
4254  012416                    ENDIF
4255  012416                                                              50201$:
4256  012416                ELSE
4257  012416  000454                                                          BR      50202$
4258  012420                                                              50170$:
4259  012420  004737  012716    JSR PC,RTLE                 ;CHECK FOR RETRY LIMIT EXCEEDED.
4260  012424                    IF CMDLG GT #2 THEN         ;IF READ CMD THEN:
4261  012424  023727  003354  000002                                          CMP     CMDLG,#2
4262  012432  003411                                                         BLE     50203$
4263  012434                      LET R2 := #RRECL SHIFT  1   ;R2=READ RETRY COUNT LIMIT / 2
4264  012434  012702  000020                                                  MOV     #RRECL,R
4265  012440  006202                                                         ASR     R2
4266  012442                      IF RETRYC GE R2 THEN              ;IF RETRY COUNT IS MORE THAN HAL
4267  012442  023702  003404                                                  CMP     RETRYC,R
4268  012446  002403                                                         BLT     50204$
4269  012450                        LET CMDPKT := CMDPKT SET.BY #OPP.C      ;SET OPPOSITE BIT FOR RE
4270  012450  052737  020000  002310                                          BIS     #OPP.C,C
4271  012456                      ENDIF
4272  012456                                                              50204$:
4273  012456                    ENDIF
4274  012456                                                              50203$:
4275  012456                  IF RETRYC EQ #0 ANDB ERCVER NE #0 THEN  ;IF THIS IS THE ORIGINAL ERROR
4276  012456  005737  003404                                                  TST     RETRYC
4277  012462  001007                                                         BNE     50205$
4278  012464  105737  002205                                                  TSTB    ERCVER
4279  012470  001404                                                         BEQ     50205$
4280  012472                      ERRSOFT #9,RERM,STAERM      ;REPORT RECOVERABLE ERROR
4281  012472  104457                                                         TRAP    C$ERSOFT
4282  012474  000011                                                         .WORD   9
4283  012476  004550                                                         .WORD   RERM
4284  012500  005372                                                         .WORD   STAERM
4285  012502                    ENDIF                       ;PROVIDED OPERATOR HAS ENABLED THE REPOR
4286  012502                                                          50205$:
4287  012502                    LET RETRYC := RETRYC + #1    ;UPDATE RETRY COUNT.
4288  012502  005237  003404                                                  INC     RETRYC
4289  012506                    LET CMDPKT := CMDPKT SET.BY #MOD.C1   ;SET RETRY BIT IN CMD PACKET.
4290  012506  052737  001000  002310                                          BIS     #MOD.C1,
4291  012514                    IFB IREC EQ #0 THEN         ;IF ERROR RECOVERY ENABLED:
4292  012514  105737  002211                                                  TSTB    IREC
4293  012520  001011                                                         BNE     50206$
4294  012522                      LET ERRREC :B- ERRREC + #1  ;SET ERROR RECOVERY FLAG.
4295  012522  105237  003415                                                  INCB    ERRREC
4296  012526                      POP R2,R2                   ;POP 2 RTN ADRS FROM STACK.
4297  012526  012602                                                         MOV     (SP)+,R2
4298  012530  012602                                                         MOV     (SP)+,R2
4299  012532  004737  010326      JSR PC,EXCUTE                   ;GO EXECUTE THE RETRY COMMAND.
```

```
 4300  012536  000137  010636                        JMP GOWAIT                ;GO WAIT FOR INTERRUPT + CHECK STATUS.
 4301  012542                                      ELSE                        ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
 4302  012542  000402                                                          BR      50207$
 4303  012544                                                              50206$:
 4304  012544                                        LET UNREC :B= UNREC + #1   ;SET UNRECOVERABLE ERROR FLAG.
 4305  012544  105237  003414                                                  INCB    UNREC
 4306  012550                                      ENDIF
 4307  012550                                                              50207$:
 4308  012550                                    ENDIF
 4309  012550                                                              50202$:
 4310  012550  000207                        3$:  RTS PC                      ;RETURN
```

```
4311                                    ;       SUBROUTINE TO HANDLE TERMINATION CLASS CODE 5, RECOVERABLE ERROR.
4312                                    ;       TAPE POSITION HAS NOT CHANGED.  RECOVERY PROCEDURE IS TO LOG THE
4313                                    ;       ERROR AND RE-ISSUE THE ORIGINAL COMMAND.
4314                                    ;       INPUTS:
4315                                    ;       OUTPUTS:
4316                                    ;       REGISTERS:      R2,R4.
4317                                    ;       CALLS:          RTLE, EXCUTE, GOWAIT, DROPU.
4318
4319  012552  004737  012716            TCC5::  JSR PC,RTLE                     ;CHECK FOR RETRY LIMIT EXCEEDED
4320  012556                            TCC5::  IF RETRYC EQ #0 THEN            ;IF THIS IS THE ORIGINAL ERROR THEN:
4321  012556  005737  003404                                                                  TST     RETRYC
4322  012562  001004                                                                          BNE     50210$
4323  012564                                    ERRSOFT #10,RERM,STAERM       ;REPORT RECOVERABLE ERROR.
4324  012564  104457                                                                          TRAP    C$ERSOFT
4325  012566  000012                                                                          .WORD   10
4326  012570  004550                                                                          .WORD   RERM
4327  012572  005372                                                                          .WORD   STAERM
4328  012574                            TCC5::  ENDIF
4329  012574                                                                  50210$:
4330  012574                            TCC5::  LET RETRYC := RETRYC + #1      ;UPDATE RETRY COUNTER.
4331  012574  005237  003404                                                                  INC     RETRYC
4332  012600                            TCC5::  IFB IREC EQ #0 THEN           ;IF ERROR RECOVERY IS ENABLED:
4333  012600  105737  002211                                                                  TSTB    IREC
4334  012604  001016                                                                          BNE     50211$
4335  012606                                    LET ERRREC :B= ERRREC + #1   ;SET ERROR RECOVERY FLAG.
4336  012606  105237  003415                                                                  INCB    ERRREC
4337  012612                                    LET RECCNT(R5) := RECCNT(R5) + #1      ;UPDATE REC COUNT
4338  012612  005265  003324                                                                  INC     RECCNT(R
4339  012616                                    LET @DATAWT := RECCNT(R5)             ;AND INSERT IT INTO WRT BFR
4340  012616  016577  003324  170510                                                          MOV     RECCNT(R
4341  012624                                    POP R2,R2                   ;POP 2 RTN ADRS FROM STACK.
4342  012624  012602                                                                          MOV     (SP)+,R2
4343  012626  012602                                                                          MOV     (SP)+,R2
4344  012630  004737  010326                    JSR PC,EXCUTE              ;GO RE ISSUE THE COMMAND.
4345  012634  000137  010636                    JMP GOWAIT                 ;GO WAIT FOR INTERRUPT + CHECK STATUS.
4346  012640                            TCC5::  ELSE                        ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
4347  012640  000402                                                                          BR      50212$
4348  012642                                                                  50211$:
4349  012642                                    LET UNREC :B- UNREC + #1     ;SET UNRECOVERABLE ERROR FLAG.
4350  012642  105237  003414                                                                  INCB    UNREC
4351  012646                            TCC5::  ENDIF
4352  012646                                                                  50212$:
4353  012646  000207                    TCC5::  RTS PC                      ;RETURN.
4354
4355
```

```
                                        ;       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:          RTIF, WSSR, EXCUTE, GOWAIT, DROPU

012650                                  TCC6::  LET @TSDB(R5) := #RWCPK         ;ISSUE A REWIND COMMAND.
012650  012775  002330  002452                                                          MOV     #RWCPK,@
012656  004737  011170                          JSR PC,WSSR                     ;WAIT FOR SUBSYSTEM READY.
012662                                          ERRDF #11,URERM,STAERM          ;REPORT UNRECOVERABLE ERROR.
012662  104455                                                                          TRAP    C$ERDF
012664  000013                                                                        .WORD   11
012666  004572                                                                        .WORD   URERM
012670  005372                                                                        .WORD   STAERM
012672  004737  015554                          JSR PC,DROPU                    ;REPORT ERROR - DROP UNIT.
012676  000207                                  RTS PC                          ;RETURN
```

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

012700                          TCC7::  ERRDF #12,FATSM,STAERM          ;REPORT FATAL SUBSYSTEM ERROR.
012700  104455                                                                  TRAP    C$ERDF
012702  000014                                                                  .WORD   12
012704  004373                                                                  .WORD   FATSM
012706  005372                                                                  .WORD   STAERM
012710  004737  015554                  JSR PC,DROPU                    ;DROP THE UNIT.
012714  000207                          RTS PC                          ;RETURN.


                                ;       SUBROUTINE TO CHECK FOR RETRY LIMIT EXCEEDED.  PRINTS ERROR MESSAGE
                                ;       IF EXCEEDED AND DROP UNIT UNLESS COMMAND IS A READ.
                                ;       INPUTS:
                                ;       OUTPUTS:
                                ;       REGISTERS:      R2, R4.
                                ;       CALLS:          DROPU

012716                          RTLE::  IF CMDLG EQ #0 THEN             ;IF CMD IS NOT A READ OR WRITE THEN:
012716  005737  003354                                                          TST     CMDLG
012722  001010                                                                BNE     50213$
012724                            ERRDF #11,URERM,STAERM                ;REPORT UNRECOVERABLE ERROR.
012724  104455                                                                  TRAP    C$ERDF
012726  000013                                                                  .WORD   11
012730  004572                                                                  .WORD   URERM
012732  005372                                                                  .WORD   STAERM
012734  004737  015554                JSR PC,DROPU                    ;DROP THE UNIT.
012740                                POP R2
012740  012602                                                                  MOV     (SP)+,R2
012742  000437                        BR RTLRTN                       ;AND RETURN.
012744                          ENDIF
012744                                                                50213$:
012744                          LET RWERR :B= RWERR + #1                ;SET READ/WRITE ERROR FLAG.
012744  105237  003413                                                          INCB    RWERR
012750                          IF CMDLG EQ #2 THEN                     ;IF CMD IS A WRT OR WTM:
012750  023727  003354  000002                                                  CMP     CMDLG,#2
012756  001016                                                                BNE     50214$
012760                            IF RETRYC EQ #WRECL THEN              ;IF RETRY COUNT HAS REACHED LIMIT:
012760  023727  003404  000020                                                  CMP     RETRYC,#
012766  001011                                                                BNE     50215$
012770                              LET UNREC :B= UNREC + #1            ;SET UNRECOVERABLE FLAG
012770  105237  003414                                                          INCB    UNREC
012774                              ERRDF #14,RLEXM,STAERM              ;REPORT RETRY LIMIT EXCEEDED.
012774  104455                                                                  TRAP    C$ERDF
012776  000016                                                                  .WORD   14
013000  004310                                                                  .WORD   RLEXM
013002  005372                                                                  .WORD   STAERM
```

```
4433  013004  004737  015554                    JSR PC,DROPU              ;DROP THE UNIT.
4434  013010                                    POP R2
4435  013010  012602                                                       MOV     (SP)+,R2
4436  013012                                  ENDIF
4437  013012                                                          50215$:
4438  013012                                ELSE                        ;ELSE - CMD IS A READ:
4439  013012  000413                                                       BR      50216$
4440  013014                                                          50214$:
4441  013014                                  IF RETRYC EQ #RRECL THEN  ;IF RETRY COUNT HAS REACHED LIMIT:
4442  013014  023727  003404  000020                                       CMP     RETRYC,#
4443  013022  001007                                                       BNE     50217$
4444  013024                                    LET UNREC :B= UNREC + #1 ;SET UNRECOVERABLE FLAG
4445  013024  105237  003414                                             INCB    UNREC
4446  013030                                    ERRHRD #14,RLEXM,STAERM  ;REPORT RECOVERABLE ERROR.
4447  013030  104456                                                     TRAP    C$ERHRD
4448  013032  000016                                                     .WORD   14
4449  013034  004310                                                     .WORD   RLEXM
4450  013036  005372                                                     .WORD   STAERM
4451  013040                                    POP R2
4452  013040  012602                                                     MOV     (SP)+,R2
4453  013042                                  ENDIF
4454  013042                                                          50217$:
4455  013042                                ENDIF
4456  013042                                                          50216$:
4457  013042  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

013044                                  WRTY::  IF DEVTBL(R5) EQ #NINUSE THEN   ;BTL
013044  026527  002532  177774                                                           CMP     DEVTBL(R
013052  001003                                                                            BNE     50220$
013054  000137  013362                          JMP 1$                                  ;BTL
013060                                          ELSE                                    ;BTL
013060  000400                                                                            BR      50221$
013062                                                                          50220$:
013062                                          ENDIF                                   ;BTL
013062                                                                          50221$:
013062                                          BEGIN RETRY
013062                                            REPEAT
013062                                                                          50223$:
013062                                              BEGIN REPEAT
013062                                                REPEAT
013062                                                                          50225$:
013062  004737  013576                            JSR PC,BORERS                   ;BACKSPACE/ERASE ONE RECORD
013066                                            LET WRTYER :B= #0               ;CLEAR WRITE RETRY ERROR
013066  105037  003410                                                                    CLRB    WRTYER
013072  004737  013752                            JSR PC,REWRT                    ;REWRITE RECORD ON SAME SPOT
013076                                             IF DEVTBL(R5) EQ #NINUSE THEN  ;BTL
013076  026527  002532  177774                                                           CMP     DEVTBL(R
013104  001003                                                                            BNE     50226$
013106  000137  013362                             JMP 1$                         ;BTL
013112                                             ELSE                           ;BTL
013112  000400                                                                            BR      50227$
013114                                                                          50226$:
013114                                             ENDIF                          ;BTL
013114                                                                          50227$:
013114                                            LET RPTCNT :B= RPTCNT + #1       ;COUNT REPEATS
013114  105237  003406                                                                    INCB    RPTCNT
013120                                          UNTILB RPTCNT EQ #4 ORB WRTYER NE #0    ;LIMIT: 4 REPEATS OR REC
013120  123727  003406  000004                                                           CMPB    RPTCNT,#
013126  001403                                                                            BEQ     50230$
013130  105737  003410                                                                    TSTB    WRTYER
013134  001752                                                                            BEQ     50225$
013136                                                                          50230$:
013136                                          END REPEAT                      ;
013136                                                                          50224$:
013136                                          LET RETRYC :- RETRYC + #1        ;COUNT RETRIES
013136  005237  003404                                                                    INC     RETRYC
```

```
4514  013142                                  IF DEVTBL(R5) EQ #NINUSE THEN       ;BTL
4515  013142  026527  002532  177774                                                    CMP     DEVTBL(R
4516  013150  001003                                                                   BNE     50231$
4517  013152  000137  013362                  JMP 1$                              ;BTL
4518  013156                                  ELSE                                ;BTL
4519  013156  000400                                                                   BR      50232$
4520  013160                                                                  50231$:
4521  013160                                  ENDIF                               ;BTL
4522  013160                                                                  50232$:
4523  013160                                  IFB WRTYER EQ #0 THEN               ;
4524  013160  105737  003410                                                           TSTB    WRTYER
4525  013164  001001                                                                   BNE     50233$
4526  013166                                    LEAVE RETRY                       ;EXIT RETRY LOOP IF RECOVERED
4527  013166  000457                                                                   BR      50222$
4528  013170                                  ELSE                                ;
4529  013170                                                                  50233$:
4530  013170                                    IFB ERCVER NE #0 THEN             ;
4531  013170  105737  002205                                                           TSTB    ERCVER
4532  013174  001415                                                                   BEQ     50235$
4533  013176                                      PRINTB #BTMSG1,RETRYC,<B,RPTCNT>      ;PRINT SUSPECTED BAD SPO
4534  013176  005046                                                                   CLR     -(SP)
4535  013200  153716  003406                                                           BISB    RPTCNT,(
4536  013204  013746  003404                                                           MOV     RETRYC,-
4537  013210  012746  013372                                                           MOV     #BTMSG1,
4538  013214  012746  000003                                                           MOV     #3,-(SP)
4539  013220  010600                                                                   MOV     SP,R0
4540  013222  104414                                                                   TRAP    C$PNTB
4541  013224  062706  000010                                                           ADD     #10,SP
4542  013230                                    ENDIF                             ;
4543  013230                                                                  50235$:
4544  013230                                    IF RETRYC EQ #1 THEN          ;ON FIRST RETRY, LOGG BAD SPOT
4545  013230  023727  003404  000001                                                   CMP     RETRYC,#
4546  013236  001021                                                                   BNE     50236$
4547  013240                                      LET BTPT := BTADDR(R5)          ;BTPT IS BOTH THE BAD SPOT COUNT
4548  013240  016537  002544  003436                                                   MOV     BTADDR(R
4549  013246                                      LET R4 := @BTPT + #2            ;AND THE LOGGING INDEX
4550  013246  017704  170164                                                           MOV     @BTPT,R4
4551  013252  062704  000002                                                           ADD     #2,R4
4552  013256                                      LET @BTPT := R4                 ;
4553  013256  010477  170154                                                           MOV     R4,@BTPT
4554  013262                                      IF R4 LOS #40. THEN             ;
4555  013262  020427  000050                                                           CMP     R4,#40.
4556  013266  101005                                                                   BHI     50237$
4557  013270                                        LET R3 := BTPT                ;STORE FIRST 20 BAD SPOTS
4558  013270  013703  003436                                                           MOV     BTPT,R3
4559  013274                                        LET R4 := R4 + R3             ;
4560  013274  060304                                                                   ADD     R3,R4
4561  013276                                        LET (R4) := RECCNT(R5)        ;
4562  013276  016514  003324                                                           MOV     RECCNT(R
4563  013302                                      ENDIF
4564  013302                                                                  50237$:
4565  013302                                  ENDIF                               ;
4566  013302                                                                  50236$:
4567  013302                                  LET ERSFLG :B= ERSFLG + #1       ;ERASE FLAG TO ERASE BAD RECORD
4568  013302  105237  003451                                                           INCB    ERSFLG
4569  013306                                  LET RWERR :B= #0                 ;CANCELL "LOG" ERROR FLAG ON FAI
```

```
4570  013306  105037  003413                                  CLRB    RWERR
4571  013312                                    LET RPTCNT :B= #0   ;CLEAR REPEAT COUNT FOR NEXT RET
4572  013312  105037  003406                                  CLRB    RPTCNT
4573  013316                                  ENDIF             ;
4574  013316                                                  50234$:
4575  013316                                UNTIL RETRYC EQ #4  ;LIMIT: 4 RETRIES
4576  013316  023727  003404  000004                          CMP     RETRYC,#
4577  013324  001256                                         BNE     50223$
4578  013326                                END RETRY           ;
4579  013326                                                  50222$:
4580  013326                                IFB WRTYER NE #0 THEN ;
4581  013326  105737  003410                                  TSTB    WRTYER
4582  013332  001413                                         BEQ     50240$
4583  013334                                  IFB ERCVER NE #0 THEN ;
4584  013334  105737  002205                                  TSTB    ERCVER
4585  013340  001410                                         BEQ     50241$
4586  013342                                    PRINTB #BTMSG3  ;PRINT RETRY FAILED
4587  013342  012746  013527                                  MOV     #BTMSG3,
4588  013346  012746  000001                                  MOV     #1,(SP)
4589  013352  010600                                         MOV     SP,R0
4590  013354  104414                                         TRAP    C$PNTB
4591  013356  062706  000004                                  ADD     #4,SP
4592  013362                                  ENDIF
4593  013362                                                  50241$:
4594  013362                                ENDIF
4595  013362                                                  50240$:
4596  013362  000207                    1$:    RTS PC
4597
4598
4599
4600
```

```
4601
4602  013364  000000                        WTYCMD: .WORD   0               ;STORAGE FOR WRITE CMD WHILE RETRYING
4603  013366  000000                        WTYWRD: .WORD   0               ;STORAGE FOR WRITE CMD WORD WHILE RETRYING
4604  013370  000000                        WTYBRF: .WORD   0               ;STORAGE FOR WRITE BPCR WHILE RETRYING
4605
4606
4607  013372  040445  052523  050123        BTMSG1: .ASCIZ  /%ASUSPECT BAD SPOT AFTER %D1%A RETRY, %D1%A REPEAT%N/
4608  013400  041505  020124  040502
4609  013406  020104  050123  052117
4610  013414  040440  052106  051105
4611  013422  022440  030504  040445
4612  013430  051040  052105  054522
4613  013436  020054  042045  022461
4614  013444  020101  042522  042520
4615  013452  052101  047045     000
4616  013457     045  022516  041101        BTMSG2: .ASCIZ  /%N%ABAD TAPE OVERFLOW: CHANGE TAPE!%N%N/
4617  013464  042101  052040  050101
4618  013472  020105  053117  051105
4619  013500  046106  053517  020072
4620  013506  044103  047101  042507
4621  013514  052040  050101  020505
4622  013522  047045  047045     000
4623  013527     045  051101  052105        BTMSG3: .ASCIZ  /%ARETRY FAILED ON BAD SPOT...ERASED!%N/
4624  013534  054522  043040  044501
4625  013542  042514  020104  047117
4626  013550  041040  042101  051440
4627  013556  047520  027124  027056
4628  013564  051105  051501  042105
4629  013572  022441  000116
4630                                                .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
                                
013576                          BORERS:: LET PCMDWD := CMDWRD   ;SET COMMAND  TO SPACE REV
013576  013737  003346  003352          MOV     CMDWRD,P
                                        LET CMDWRD := #SRR      ;
013604  012737  104410  003346          MOV     #SRR,CMD
                                        LET CMDPKT := CMDWRD CLR.BY #BRF.C      ;
013612  013737  003346  002310          MOV     CMDWRD,C
013620  042737  004000  002310          BIC     #BRF.C,C
                                        LET CMDSAV := CMDPKT    ;
013626  013737  002310  003350          MOV     CMDPKT,C
                                        LET CMDPKT+CP.ADL := #1 ;
013634  012737  000001  002312          MOV     #1,CMDPK
                                        LET CMDLG := #0         ;
013642  005037  003354                  CLR     CMDLG
013646  004737  007344                  JSR PC,CMDAC            ;
013652  004737  010326                  JSR PC,EXCUTE           ;
013656  004737  010636                  JSR PC,GOWAIT           ;
013662  004737  016060                  JSR PC,CKHAE            ;
013666                                  IFB ERSFLG NE #0 THEN   ;WHEN ERASE FLAG IS SET, DO ERASE
013666  105737  003451                          TSTB    ERSFLG
013672  001426                                  BEQ     50242$
013674                                    LET PCMDWD := CMDWRD  ;
013674  013737  003346  003352                  MOV     CMDWRD,P
                                          LET CMDWRD := #ERS    ;
013702  012737  100411  003346                  MOV     #ERS,CMD
                                          LET CMDPKT := CMDWRD  ;
013710  013737  003346  002310                  MOV     CMDWRD,C
                                            LET CMDSAV := CMDPKT  ;
013716  013737  002310  003350                  MOV     CMDPKT,C
013724  004737  007344                    JSR PC,CMDAC            ;
013730  004737  010326                    JSR PC,EXCUTE
013734  004737  010636                    JSR PC,GOWAIT
013740  004737  016060                    JSR PC,CKHAE
013744                                    LET ERSFLG :B= #0
013744  105037  003451                          CLRB    ERSFLG
013750                                  ENDIF
013750                          50242$:
013750  000207                          RTS PC
                                ;       SUBR TO REWRITE A BADLY WRITTEN RECORD
013752                          REWRT:  IF DEVTBL(R5) EQ #NINUSE THEN   ;BTL
013752  026527  002532  177774                  CMP     DEVTBL(R
013760  001003                                  BNE     50243$
013762  000137  014100                  JMP 1$                  ;BTL
013766                                  ELSE                    ;BTL
013766  000400                                  BR      50244$
013770                          50243$:
013770                                  ENDIF                   ;BTL
013770                          50244$:
013770                          LET PCMDWD := CMDWRD    ;RESTORE WRITE COMMAND PACKET
```

```
4687  013770  013737  003346  003352
4688  013776
4689  013776  013737  013366  003346
4690  014004
4691  014004  013737  013364  002310
4692  014012
4693  014012  013737  002310  003350
4694  014020
4695  014020  013737  003334  002312
4696  014026
4697  014026  013737  013370  002316
4698  014034
4699  014034  012737  000002  003354
4700  014042  004737  007344
4701  014046  004737  010326
4702  014052
4703  014052  026527  002532  177774
4704  014060  001003
4705  014062  000137  014100
4706  014066
4707  014066  000400
4708  014070
4709  014070
4710  014070
4711  014070  004737  010636
4712  014074  004737  016060
4713  014100  000207
```

```
                LET CMDWRD := WTYWRD     ;
                LET CMDPKT := WTYCMD     ;
                LET CMDSAV := CMDPKT     ;
                LET CMDPKT+CP.ADL := DATAWT     ;
                LET CMDPKT+CP.CNT := WTYBRF     ;
                LET CMDLG := #2          ;
                JSR PC,CMDAC
                JSR PC,EXCUTE            ;RE WRITE RECORD
                IF DEVTBL(R5) EQ #NINUSE THEN   ;BTL
                JMP 1$                   ;BTL
                ELSE                     ;BTL
                ENDIF                    ;BTL
                JSR PC,GOWAIT            ;
                JSR PC,CKHAE             ;
1$:             RTS PC
```

```
                                                SEQ 0113
                MOV     CMDWRD,P
                MOV     WTYWRD,C
                MOV     WTYCMD,C
                MOV     CMDPKT,C
                MOV     DATAWT,C
                MOV     WTYBRF,C
                MOV     #2,CMDLG

                CMP     DEVTBL(R
                BNE     50245$

                BR      50246$
50245$:
50246$:
```

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

014102                          LOG::   IFB ERLOG EQ #0 THEN            ;IF DATA AND ERRORS HAVE NOT BEEN LOGGED
014102  105737  003412                                                            TSTB    ERLOG
014106  001126                                                                    BNE     50247$
014110                                  LET ERLOG :B= ERLOG + #1         ;SET LOG DONE FLAG.
014110  105237  003412                                                            INCB    ERLOG
014114                                  LET R4 := CMDLG                  ;GET CURRENT CMD LOGGING CODE.
014114  013704  003354                                                            MOV     CMDLG,R4
014120                                  IF R4 NE #0 THEN                 ;IF THERE IS A CODE THEN:
014120  005704                                                                    TST     R4
014122  001520                                                                    BEQ     50250$
014124                                    LET R4 := R4   #2              ;ADJUST THE CODE FOR TABLE INDEX.
014124  162704  000002                                                            SUB     #2,R4
014130                                    LET R2 := R5 + BINC(R4) + #CNTBGN  ;R2 - ADR OF BYTE COUNT LSW.
014130  010502                                                                    MOV     R5,R2
014132  066402  014366                                                            ADD     BINC(R4)
014136  062702  002554                                                            ADD     #CNTBGN,
014142                                    LET (R2) := (R2) + BRFCNT      ;ADD BRF TO LSW.
014142  063712  003344                                                            ADD     BRFCNT,(
014146                                    IF MSGPKT+MS.RFC LOS BRFCNT THEN  ;IF THE RFC IS LOWER OR THE SAME AS
014146  023737  002340  003344                                                    CMP     MSGPKT+M
014154  101002                                                                    BHI     50251$
014156                                      LET (R2) := (R2) - MSGPKT+MS.RFC  ;SUBTRACT RFC FROM EXPECTED BRF.
014156  163712  002340                                                            SUB     MSGPKT+M
014162                                    ENDIF
014162                                                                  50251$:
014162                                  LET R3 := R2 + #10               ;R3 = ADR OF 2ND WORD.
014162  010203                                                                    MOV     R2,R3
014164  062703  000010                                                            ADD     #10,R3
014170                                  WHILE (R2) GT #999. DO
014170                                                                  50252$:
014170  021227  001747                                                            CMP     (R2),#99
014174  003404                                                                    BLE     50253$
014176                                    LET (R2) := (R2)   #1000.    ;UPDATE BYTE COUNT
014176  162712  001750                                                            SUB     #1000.,(
014202                                    LET (R3) := (R3) + #1          ;2ND WORD.
014202  005213                                                                    INC     (R3)
014204                                  ENDDO
014204  000771                                                                    BR      50252$
014206                                                                  50253$:
014206                                  LET R2 := R3 + #10               ;R2 = ADR OF 3RD WORD.
014206  010302                                                                    MOV     R3,R2
014210  062702  000010                                                            ADD     #10,R2
014214                                  WHILE (R3) GT #999. DO
014214                                                                  50254$:
014214  021327  001747                                                            CMP     (R3),#99
014220  003404                                                                    BLE     50255$
014222                                    LET (R3) := (R3)   #1000.    ;UPDATE BYTE COUNT
014222  162713  001750                                                            SUB     #1000.,(
014226                                    LET (R2) := (R2) + #1          ;3RD WORD.
```

```
4770  014226  005212                                                    INC     (R2)
4771  014230                                ENDDO
4772  014230  000771                                                    BR      50254$
4773  014232                                                          50255$:
4774  014232                                LET R3 := R2 + #10          ;R3 = ADR OF 4TH WORD.
4775  014232  010203                                                    MOV     R2,R3
4776  014234  062703  000010                                           ADD     #10,R3
4777  014240                                WHILE (R2) GT #999. DO
4778  014240                                                          50256$:
4779  014240  021227  001747                                           CMP     (R2),#99
4780  014244  003404                                                    BLE     50257$
4781  014246                                  LET (R2) :- (R2)   #1000.   ;UPDATE BYTE COUNT
4782  014246  162712  001750                                           SUB     #1000.,(
4783  014252                                  LET (R3) := (R3) + #1       ;4TH WORD.
4784  014252  005213                                                    INC     (R3)
4785  014254                                ENDDO
4786  014254  000771                                                    BR      50256$
4787  014256                                                          50257$:
4788  014256                                IFB RWERR NE #0 THEN        ;IF R/W ERROR, UPDATE ERROR COUNT.
4789  014256  105737  003413                                           TSTB    RWERR
4790  014262  001440                                                    BEQ     50260$
4791  014264                                  LET R2 := R5 + EINC(R4) + #WRREC  ;R2 = ADR OF COUNTER.
4792  014264  010502                                                    MOV     R5,R2
4793  014266  066402  014374                                           ADD     EINC(R4)
4794  014272  062702  002714                                           ADD     #WRREC,R
4795  014276                                  IFB UNREC NE #0 THEN        ;IS THE ERROR UNRECOVERABLE?
4796  014276  105737  003414                                           TSTB    UNREC
4797  014302  001404                                                    BEQ     50261$
4798  014304                                    LET R2 := R2 + #10        ;YES, POINT TO NEXT COUNTER.
4799  014304  062702  000010                                           ADD     #10,R2
4800  014310                                    LET (R2) := (R2) + #1     ;UPDATE THE ERROR COUNTER
4801  014310  005212                                                    INC     (R2)
4802  014312                                ELSE                        ;ELSE   IF ERROR IS RECOVERABLE:
4803  014312  000424                                                    BR      50262$
4804  014314                                                          50261$:
4805  014314                                  LET (R2) := (R2) + #1     ;UPDATE THE ERROR COUNTER
4806  014314  005212                                                    INC     (R2)
4807  014316                                  IFB IREC EQ #0 THEN       ;IF ERROR RECOVERY IS ENABLED:
4808  014316  105737  002211                                           TSTB    IREC
4809  014322  001020                                                    BNE     50263$
4810  014324                                    IFB DROPED EQ #0 ANDB ERCVER NE #0 THEN ;IF UNIT HAS NOT BEEN DR
4811  014324  105737  003446                                           TSTB    DROPED
4812  014330  001015                                                    BNE     50264$
4813  014332  105737  002205                                           TSTB    ERCVER
4814  014336  001412                                                    BEQ     50264$
4815  014340                                      PRINTB #NURTY1,RETRYC     ;PRINT # OF RETRIES TO RECOVER
4816  014340  013746  003404                                           MOV     RETRYC,
4817  014344  012746  005073                                           MOV     #NURTY1,
4818  014350  012746  000002                                           MOV     #2,(SP)
4819  014354  010600                                                    MOV     SP,R0
4820  014356  104414                                                    TRAP    C$PNTB
4821  014360  062706  000006                                           ADD     #6,SP
4822  014364                                    ENDIF                     ;PROVIDED PRINT HAS BEEN ENABLED
4823  014364                                                          50264$:
4824  014364                                  ENDIF
4825  014364                                                          50263$:
```

```
 4826  014364                                          ENDIF
 4827  014364                                                                  50262$:
 4828  014364                                        ENDIF
 4829  014364                                                                  50260$:
 4830  014364                                      ENDIF
 4831  014364                                                                  50250$:
 4832  014364                                      ENDIF
 4833  014364                                                                  50247$:
 4834  014364  000207                              RTS PC
 4835                                  ;           INDEXES TO BYTE COUNTERS.
 4836  014366  000000                  BINC:       0                           ;WRITE.
 4837  014370  000040                              40                          ;READ REV.
 4838  014372  000100                              100                         ;READ FWD.
 4839                                  ;           INDEXES TO READ/WRITE ERROR COUNTERS.
 4840  014374  000000                  EINC:       0                           ;WRITE.
 4841  014376  000020                              20                          ;READ REV.
 4842  014400  000040                              40                          ;READ FWD.
 4843
 4844
```

```
                                        ;       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:
                                        ;       REGISTERS:
                                        ;       CALLS:          VFEXC.

014402                                  VFYDAT::  IFB VFYFLG NE #0 THEN         ;IF DATA IS TO BE VERIFIED:
014402  105737  003442                                                                    TSTB    VFYFLG
014406  001435                                                                           BEQ     50265$
014410                                            LET PCMDWD := CMDWRD          ;SAVE THE PREVIOUS COMMAND WORD.
014410  013737  003346  003352                                                           MOV     CMDWRD,P
014416                                            LET CMDWRD :- #RDR            ;COMMAND IS READ REV.
014416  012737  104401  003346                                                           MOV     #RDR,CMD
014424                                            LET CMDLG := #4               ;SET UP CMD LOGGING INDEX.
014424  012737  000004  003354                                                           MOV     #4,CMDLG
014432  004737  014504                           JSR   PC,VFEXC                ;GO READ ALL THE RECORDS REV.
014436                                            IF DEVTBL(R5) EQ #NINUSE THEN ;BTL
014436  026527  002532  177774                                                           CMP     DEVTBL(R
014444  001003                                                                         BNE     50266$
014446  000137  014502                           JMP 1$                        ;BTL
014452                                            ELSE                          ;BTL
014452  000400                                                                         BR      50267$
014454                                                                        50266$:
014454                                            ENDIF                         ;BTL
014454                                                                        50267$:
014454                                            LET PCMDWD := CMDWRD          ;SAVE THE PREVIOUS COMMAND WORD.
014454  013737  003346  003352                                                           MOV     CMDWRD,P
014462                                            LET CMDWRD := #RDF            ;COMMAND IS READ FWD.
014462  012737  104001  003346                                                           MOV     #RDF,CMD
014470                                              LET CMDLG :  #6             ;SET UP CMD LOGGING INDEX.
014470  012737  000006  003354                                                           MOV     #6,CMDLG
014476  004737  014504                             JSR   PC,VFEXC              ;GO READ ALL RECORDS FWD.
014502                                            ENDIF
014502                                                                        50265$:
014502  000207                          1$:       RTS     PC          ;RETURN.
```

```
4883                                    ;       SUBROUTINE TO EXECUTE THE READ AND VERIFY, FORWARD OR REVERSE.
4884                                    ;       INPUTS:
4885                                    ;       OUTPUTS:
4886                                    ;       REGISTERS:      R2
4887                                    ;       CALLS:          CMDAC, FIRSTU, VFISU, NEXTU, CKHAE.
4888
4889  014504                            VFEXC:: LET CMDPKT := CMDWRD CLR.BY @BRF.C  ;COMMAND PACKET = READ REV OR FWD.
4890  014504  013737  003346  002310                                                      MOV     CMDWRD,C
4891  014512  042737  004000  002310                                                      BIC     @BRF.C,C
4892  014520                                    IFB SWBFLG NE @0 THEN           ;IF BYTES ARE TO BE SWAPPED:
4893  014520  105737  003444                                                            TSTB    SWBFLG
4894  014524  001403                                                                    BEQ     50270$
4895  014526                                      LET CMDPKT := CMDPKT SET.BY @SWB.C  ;SET SWAB BIT IN CMD PACKET.
4896  014526  052737  010000  002310                                                    BIS     @SWB.C,C
4897  014534                                    ENDIF
4898  014534                                                                         50270$:
4899  014534                                    LET CMDSAV := CMDPKT            ;SAVE COMMAND PACKET 1ST WORD.
4900  014534  013737  002310  003350                                                    MOV     CMDPKT,C
4901  014542  013737  003336  002312          MOV     DATARD,CMDPKT+CP.ADL    ;SAVE BUFFER START ADDRESS.
4902  014550                                  LET NCNT := @0                  ;CLEAR NUMBER OF OPERATIONS.
4903  014550  005037  003340                                                          CLR     NCNT
4904  014554                                  WHILE NCNT LT NCNT1 DO          ;WHILE THERE ARE RECORDS REMAINING:
4905  014554                                                                         50271$:
4906  014554  023737  003340  003342                                                  CMP     NCNT,NCN
4907  014562  002071                                                                 BGE     50272$
4908  014564  004737  007344                    JSR PC,CMDAC                  ;STORE CMD ASCII IN ERROR MSG.
4909  014570  004737  015452                    JSR   PC,FIRSTU               ;SET UP FOR FIRST UNIT.
4910  014574                                    WHILE DEVTBL(R5) NE @END DO   ;WHILE THERE ARE DEVICES REMAINING:
4911  014574                                                                         50273$:
4912  014574  026527  002532  177777                                                  CMP     DEVTBL(R
4913  014602  001442                                                                 BEQ     50274$
4914  014604                                      IF @MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
4915  014604  032737  000400  003346                                                  BIT     @MOD.CO,
4916  014612  001421                                                                 BEQ     50275$
4917  014614                                    IF @X0.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT
4918  014614  032765  000002  003426                                                  BIT     @X0.BOT,
4919  014622  001014                                                                 BNE     50276$
4920  014624                                      IF @X0.EOT SETIN EOTFLG(R5) THEN   ;BUT IF AT EOT
4921  014624  032765  000001  003426                                                  BIT     @X0.EOT,
4922  014632  001406                                                                 BEQ     50277$
4923  014634                                        IFB ALLEOT NE @0 THEN           ;AND ALL OTHERS AT EOT
4924  014634  105737  003450                                                          TSTB    ALLEOT
4925  014640  001402                                                                 BEQ     50300$
4926  014642  004737  014750                      JSR PC,VFISU                  ;THEN READ VERIFY
4927  014646                                      ENDIF                         ;IF NOT ALL AT EOT, FREEZE
4928  014646                                                                       50300$:
4929  014646                                    ELSE                            ;IF NOT AT BOT AND
4930  014646  000402                                                                 BR      50301$
4931  014650                                                                       50277$:
4932  014650  004737  014750                      JSR PC,VFISU                  ;NOT AT EOT, READ VFY
4933  014654                                    ENDIF
4934  014654                                                                       50301$:
4935  014654                                  ENDIF
4936  014654                                                                       50276$:
4937  014654                                ELSE                        ;ELSE IF CMD IS NOT REVERSE:
4938  014654  000412                                                                 BR      50302$
```

```
4939  014656                                                                                    50275$:
4940  014656                                  IF @XO.EOT NOTSETIN FOTFLG(R5) OR @CMD.CO NOTSETIN CMDWRD THEN
4941  014656  032765  000001  003426                                                            BIT     @XO.EOT,
4942  014664  001404                                                                            BEQ     50303$
4943  014666  032737  000001  003346                                                            BIT     @CMD.CO,
4944  014674  001002                                                                            BNE     50304$
4945  014676                                                                                    50303$:
4946                                          ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
4947  014676  004737  014750                   JSR PC,VFISU          ;ISSUE CMD, CHECK STATUS AND DATA.
4948  014702                                  ENDIF
4949  014702                                                                                    50304$:
4950  014702                                 ENDIF
4951  014702                                                                                    50302$:
4952  014702  004737  015520                 JSR PC,NEXTU           ;GO FIND THE NEXT UNIT.
4953  014706                                ENDDO
4954  014706  000732                                                                            BR      50273$
4955  014710                                                                                    50274$:
4956  014710  004737  016060                 JSR   PC,CKHAE         ;CHECK FOR HALT AFTER EACH CMD.
4957  014714                                  IF DEVTBL(R5) EQ @NINUSE THEN ;BTL
4958  014714  026527  002532  177774                                                            CMP     DEVTBL(R
4959  014722  001003                                                                            BNE     50305$
4960  014724  000137  014746                 JMP 1$                 ;BTL
4961  014730                                  ELSE                   ;BTL
4962  014730  000400                                                                            BR      50306$
4963  014732                                                                                    50305$:
4964  014732                                  ENDIF                  ;BTL
4965  014732                                                                                    50306$:
4966  014732                                  LET NCNT := NCNT + #1  ;UPDATE THE RECORD COUNT.
4967  014732  005237  003340                                                                    INC     NCNT
4968  014736                                  LET PCMDWD := CMDWRD   ;SAVE PREVIOUS COMMAND WORD.
4969  014736  013737  003346  003352                                                            MOV     CMDWRD,P
4970  014744                                ENDDO
4971  014744  000703                                                                            BR      50271$
4972  014746                                                                                    50272$:
4973  014746  000207                  1$:     RTS     PC             ;RETURN.
```

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

014750                                  VFISU::     LET R2 := DATARD + 08.      ;INIT READ BUFFER POINTER.
014750  013702  003336                                                                    MOV     DATARD,R
014754  062702  000010                                                                    ADD     #8.,R2
014760                                              WHILE R2 NE DATARD DO       ;UNTIL 8 BYTES HAVE BEEN SET.
014760                                                                                  50307$:
014760  020237  003336                                                                    CMP     R2,DATAR
014764  001403                                                                          BEQ     50310$
014766                                                LET  (R2) := # 1          ;INIT READ BUFFER.
014766  012742  177777                                                                    MOV     # 1, (R2
014772                                              ENDDO
014772  000772                                                                          BR      50307$
014774                                                                              50310$:
014774  004737  010326                              JSR PC,EXCUTE               ;GO EXECUTE THE COMMAND.
015000                                              IFB DROPED EQ #0 THEN       ;IF UNIT HAS NOT BEEN DROPPED THEN:
015000  105737  003446                                                                    TSTB    DROPED
015004  001002                                                                          BNE     50311$
015006  004737  010636                                JSR     PC,GOWAIT           ;GO WAIT FOR DONE BIT.
015012                                              ENDIF
015012                                                                              50311$:
015012                                              IFB DROPED EQ #0 THEN       ;IF UNIT HAS NOT BEEN DROPPED THEN:
015012  105737  003446                                                                    TSTB    DROPED
015016  001006                                                                          BNE     50312$
015020                                                IF #X0.BOT NOTSETIN EOTFLG(R5) THEN     ;WHEN NOT REVERSED INTO B
015020  032765  000002  003426                                                            BIT     #X0.BOT,
015026  001002                                                                          BNE     50313$
015030  004737  015036                                  JSR     PC,CKDATA       ;GO VERIFY DATA.
015034                                                ENDIF
015034                                                                              50313$:
015034                                              ENDIF
015034                                                                              50312$:
015034  000207                                      RTS PC
```

```
5013                            ;       SUBROUTINE TO COMPARE DATA BETWEEN READ AND WRITE BUFFERS
5014                            ;       AND PRINT ERROR MESSAGE ON MISCOMPARE.
5015                            ;       INPUTS:
5016                            ;       OUTPUTS:
5017                            ;       REGISTERS:      R2, R3, R4.
5018                            ;       CALLS:          GCMDA
5019
5020  015036                    CKDATA:: LET R3 := BRFCNT   MSGPKT+MS.RFC ;COMPUTE REC LENGTH READ
5021  015036  013703  003344                                                        MOV     BRFCNT,R
5022  015042  163703  002340                                                        SUB     MSGPKT+M
5023  015046                            IF R3 EQ #0 THEN                ;WHEN NO DATA RECEIVED
5024  015046  005703                                                                TST     R3
5025  015050  001015                                                                BNE     50314$
5026  015052                              ERRHRD 17,WTVERM,DTAERM               ;PRINT ERROR AND EXIT
5027  015052  104456                                                                TRAP    C$ERHRD
5028  015054  000021                                                                .WORD   17
5029  015056  004164                                                                .WORD   WTVERM
5030  015060  005224                                                                .WORD   DTAERM
5031  015062                              PRINTB #DTAER4              ;COMPARE ROUTINE
5032  015062  012746  005010                                                        MOV     #DTAER4,
5033  015066  012746  000001                                                        MOV     #1, (SP)
5034  015072  010600                                                                MOV     SP,R0
5035  015074  104414                                                                TRAP    C$PNTB
5036  015076  062706  000004                                                        ADD     #4,SP
5037  015102                            ELSE
5038  015102  000560                                                                BR      50315$
5039  015104                                                          50314$:
5040  015104                              IF R3 HI BRFCNT THEN          ;WHEN REC READ IS LONGER
5041  015104  020337  003344                                                        CMP     R3,BRFCN
5042  015110  101417                                                                BLOS    50316$
5043  015112                                ERRHRD 17,WTVERM,DTAERM           ;THAN EXPECTED, PRINT
5044  015112  104456                                                                TRAP    C$ERHRD
5045  015114  000021                                                                .WORD   17
5046  015116  004164                                                                .WORD   WTVERM
5047  015120  005224                                                                .WORD   DTAERM
5048  015122                              PRINTB #DTAER5,CMDPKT+CP.CNT      ;AN ERROR MESSAGE
5049  015122  013746  002316                                                        MOV     CMDPKT+C
5050  015126  012746  005031                                                        MOV     #DTAER5,
5051  015132  012746  000002                                                        MOV     #2,-(SP)
5052  015136  010600                                                                MOV     SP,R0
5053  015140  104414                                                                TRAP    C$PNTB
5054  015142  062706  000006                                                        ADD     #6,SP
5055  015146                            ELSE                            ;AND EXIT ROUTINE
5056  015146  000536                                                                BR      50317$
5057  015150                                                          50316$:
5058  015150                              LET CKDCNT := R3   #1              ;SAVE VERIFICATION LENGTH   1.
5059  015150  010337  015446                                                        MOV     R3,CKDCN
5060  015154  005337  015446                                                        DEC     CKDCNT
5061  015160  005037  015450             CLR CKDFF                       ;CLEAR # OF BYTES IN ERROR COUNTER.
5062  015164  005002                      CLR R2                          ;INIT BYTE COUNTER
5063  015166                              LET R3 := DATAWT                ;GET WRITE BUFFER ADDRESS.
5064  015166  013703  003334                                                        MOV     DATAWT,R
5065  015172                              LET R4 := DATARD                ;GET READ BUFFER ADDRESS.
5066  015172  013704  003336                                                        MOV     DATARD,R
5067  015176                              IFB T1SWB NE #0 THEN            ;WHEN RUNNING TEST1 SUB 12.
5068  015176  105737  003447                                                        TSTB    T1SWB
```

```
5069  015202  001401                                                                  BEQ     50320$
5070  015204  000313                                  SWAB (R3)                       ;SWAP FIRST WORD OF WRT BFR
5071  015206                                          ENDIF                           ;WHICH CONTAINS THE RECORD COUNT
5072  015206                                                                          50320$:
5073  015206                                          REPEAT                          ;REPEAT UNTIL ALL DATA IS COMPARED:
5074  015206                                                                          50321$:
5075  015206                                            IF R2 EQ CKDCNT THEN          ;IF THIS IS THE LAST BYTE THEN:
5076  015206  020237  015446                                                          CMP     R2,CKDCNT
5077  015212  001011                                                                  BNE     50322$
5078  015214                                              IFB SWBFLG NE #0 THEN       ;IF BYTE SWAPPING IS ENABLED THEN:
5079  015214  105737  003444                                                          TSTB    SWBFLG
5080  015220  001406                                                                  BEQ     50323$
5081  015222                                                IF #BIT00 NOTSETIN CKDCNT THEN        ;IF RECORD LENGTH IS ODD
5082  015222  032737  000001  015446                                                  BIT     #BIT00,C
5083  015230  001002                                                                  BNE     50324$
5084  015232  105723                                               TSTB    (R3)+     ;LAST BYTE WILL BE IN
5085  015234  105724                                               TSTB    (R4)+     ;THE UPPER BYTE.
5086  015236                                                      ENDIF
5087  015236                                                                          50324$:
5088  015236                                              ENDIF
5089  015236                                                                          50323$:
5090  015236                                            ENDIF
5091  015236                                                                          50322$:
5092  015236  121314                                    CMPB (R3),(R4)              ;ARE THEY EQUAL.
5093  015240  001452                                    BEQ     3$                  ;BR IF SO.
5094  015242  005737  015450                            TST CKDFF                   ;1 ST TIME THRU?
5095  015246  001010                                    BNE     2$                  ;BR IF NOT.
5096  015250  005265  003274                            INC     VFYCNT(R5)          ;INC THE VERIFY ERROR COUNTER.
5097  015254  005265  003304                            INC HRDCNT(R5)              ;INC THE HARD ERROR COUNT.
5098  015260                                            ERRHRD #17,WTVERM,DTAERM    ;REPORT WRITE/VERIFY ERROR.
5099  015260  104456                                                                  TRAP    C$ERHRD
5100  015262  000021                                                                  .WORD   17
5101  015264  004164                                                                  .WORD   WTVERM
5102  015266  005224                                                                  .WORD   DTAERM
5103  015270                                  2$:       LET CKDFF := CKDFF + #1      ;INCREMENT # OF BYTES IN ERROR.
5104  015270  005237  015450                                                          INC     CKDFF
5105  015274  111437  003364                            MOVB (R4),TIME1             ;SAVE WAS DATA FOR TYPOUT.
5106  015300  042737  177400  003364                    BIC     #177400,TIME1       ;CLEAR GARBAGE.
5107  015306  111337  003366                            MOVB (R3),TIME2             ;SAVE SHOULD BE DATA FOR TYPOUT.
5108  015312  042737  177400  003366                    BIC     #177400,TIME2       ;CLEAR GARBAGE.
5109  015320                                            IF CKDFF LT #11. THEN       ;IF ERROR BYTE COUNT IS LESS THAN 11:
5110  015320  023727  015450  000013                                                  CMP     CKDFF,#1
5111  015326  002017                                                                  BGE     50325$
5112  015330                                              PRINTX #DTAER2,R2,<B,TIME1>,<B,TIME2>  ;PRINT EXP . ACT DATA.
5113  015330  005046                                                                  CLR     -(SP)
5114  015332  153716  003366                                                          BISB    TIME2,(S
5115  015336  005046                                                                  CLR     -(SP)
5116  015340  153716  003364                                                          BISB    TIME1,(S
5117  015344  010246                                                                  MOV     R2,-(SP)
5118  015346  012746  004677                                                          MOV     #DTAER2,
5119  015352  012746  000004                                                          MOV     #4,-(SP)
5120  015356  010600                                                                  MOV     SP,R0
5121  015360  104415                                                                  TRAP    C$PNTX
5122  015362  062706  000012                                                          ADD     #12,SP
5123  015366                                            ENDIF
5124  015366                                                                          50325$:
```

```
5125  015366  105723                3$:       TSTB (R3)+                      ;UPDATE WRITE BUFFER ADDRESS.
5126  015370  105724                          TSTB (R4)+                      ;UPDATE READ BUFFER ADDRESS.
5127  015372  105722                          TSTB (R2)+                      ;UPDATE BYTE COUNTER.
5128  015374                                UNTIL R2 GT CKDCNT          ;END OF DATA COMPARE REPEAT LOOP.
5129  015374  020237  015446                                                    CMP     R2,CKDCN
5130  015400  003702                                                          BLE     50321$
5131  015402                                LET CKDCNT := CKDCNT + #1   ;CKDCNT EQUALS RECORD LENGTH.
5132  015402  005237  015446                                                    INC     CKDCNT
5133  015406                                IF CKDFF NE #0 THEN         ;IF COMPARE ERROR HAS OCCURED THEN:
5134  015406  005737  015450                                                    TST     CKDFF
5135  015412  001414                                                          BEQ     50326$
5136  015414                                  PRINTB #DTAER3,CKDFF,CKDCNT         ;PRINT # OF BYTES IN ERROR.
5137  015414  013746  015446                                                    MOV     CKDCNT,-(
5138  015420  013746  015450                                                    MOV     CKDFF,-(
5139  015424  012746  004746                                                    MOV     #DTAER3,
5140  015430  012746  000003                                                    MOV     #3,(SP)
5141  015434  010600                                                          MOV     SP,R0
5142  015436  104414                                                          TRAP    C$PNTB
5143  015440  062706  000010                                                    ADD     #10,SP
5144  015444                                  ENDIF
5145  015444                                                              50326$:
5146  015444                                ENDIF
5147  015444                                                              50317$:
5148  015444                              ENDIF
5149  015444                                                              50315$:
5150  015444  000207                      RTS     PC                  ;OTHERWISE, RETURN.
5151
5152  015446  000000              CKDCNT: .WORD   0                   ;# OF BYTES TO BE VERIFIED  1.
5153  015450  000000              CKDFF:  .WORD   0                   ;# OF BYTES IN ERROR COUNTER.
```

```
5154                                    ;       SUBROUTINE TO FIND THE FIRST DEVICE IN THE TEST SEQUENCE.
5155                                    ;       INPUTS:
5156                                    ;       OUTPUTS:
5157                                    ;       REGISTERS:
5158                                    ;       CALLS:
5159
5160  015452                            FIRSTU:: LET DROPED :B= #0              ;CLR UNIT DROPPED FLAG
5161  015452  105037  003446                                                              CLRB    DROPED
5162  015456                                    LET R5 := #0                    ;CLR DEVICE PO  TER.
5163  015456  005005                                                                     CLR     R5
5164  015460                                    WHILE DEVTBL(R5) EQ #NINUSE DO  ;WHILE DEVICES ARE NOT IN USE:
5165  015460                                                                                  50327$:
5166  015460  026527  002532  177774                                                       CMP     DEVTBL(R
5167  015466  001003                                                                     BNE     50330$
5168  015470                                      LET R5 := R5 + #2             ;POINT TO NEXT DEVICE.
5169  015470  062705  000002                                                             ADD     #2,R5
5170  015474                                    ENDDO
5171  015474  000771                                                                     BR      50327$
5172  015476                                                                          50330$:
5173  015476                                    IF DEVTBL(R5) EQ #END THEN      ;IF ALL UNITS HAVE BEEN DROPPED THEN:
5174  015476  026527  002532  177777                                                       CMP     DEVTBL(R
5175  015504  001001                                                                     BNE     50331$
5176  015506                                      DOCLN                           ;DO CLEAN CODE AND TERMINATE PASS.
5177  015506  104444                                                                     TRAP    C$DCLN
5178  015510                                    ENDIF
5179  015510                                                                          50331$:
5180  015510                                    LET L$LUN := DEVTBL(R5)         ;SET UNIT # IN "HEADER" FOR ERROR REPOR
5181  015510  016537  002532  002074                                                       MOV     DEVTBL(R
5182  015516  000207                                    RTS     PC              ;RETURN WITH 1ST DEVICE IN R5.
5183
5184
5185
5186
5187
5188                                    ;       SUBROUTINE TO FIND THE NEXT UNIT IN THE TEST CYCLE.
5189                                    ;       INPUTS:
5190                                    ;       OUTPUTS:
5191                                    ;       REGISTERS:
5192                                    ;       CALLS:
5193
5194  015520                            NEXTU:: LET DROPED :B= #0               ;CLR UNIT DROPPED FLAG
5195  015520  105037  003446                                                              CLRB    DROPED
5196  015524  042705  177770                                    BIC     #177770,R5      ;BTL
5197  015530                                    REPEAT                          ;REPEAT UNTIL THE NEXT DEVICE IS FOUND.
5198  015530                                                                          50332$:
5199  015530                                      LET R5 := R5 + #2             ;UPDATE DEVICE TABLE POINTER.
5200  015530  062705  000002                                                             ADD     #2,R5
5201  015534                                    UNTIL DEVTBL(R5) NE #NINUSE
5202  015534  026527  002532  177774                                                       CMP     DEVTBL(R
5203  015542  001772                                                                     BEQ     50332$
5204  015544                                    LET L$LUN := DEVTBL(R5)         ;SET UNIT # IN  HEADER  FOR ERROR REPORT
5205  015544  016537  002532  002074                                                       MOV     DEVTBL(R
5206  015552  000207                                    RTS     PC              ;RETURN.
5207
5208
5209
```

```
5210                                    ;       SUBROUTINE TO DROP A DEVICE FROM THE TEST SEQUENCE.
5211                                    ;       INPUTS:
5212                                    ;       OUTPUTS:
5213                                    ;       REGISTERS:
5214                                    ;       CALLS:          MOVMSG, PRXST, LOG
5215
5216  015554                            DROPU:: LET R5 := R5SAVE                ;BTL
5217  015554  013705  003400                                                            MOV     R5SAVE,R
5218  015560                                    LET FTLCNT(R5) := FTLCNT(R5) + #1  ;INCREMENT THE FATAL ERROR COUNT.
5219  015560  005265  003314                                                            INC     FTLCNT(R
5220  015564                                    LET R4 := MSGPKT+MS.XS3 CLR.BY #377 ;GET UDIAG ERROR CODE FROM XSTAT3.
5221  015564  013704  002350                                                            MOV     MSGPKT+M
5222  015570  042704  000377                                                            BIC     #377,R4
5223  015574                                    LET R3 := MSGPKA(R5)              ;ADR OF THIS UNIT'S MSG PACKET.
5224  015574  016503  002502                                                            MOV     MSGPKA(R
5225  015600                                    LET R2 := #0                      ;CLR COUNTER.
5226  015600  005002                                                                    CLR     R2
5227  015602                                    WHILE R2 NE #MSGCNT DO            ;WHILE THERE ARE MORE LOCATIONS:
5228  015602                                                                    50333$:
5229  015602  020227  000016                                                            CMP     R2,#MSGC
5230  015606  001405                                                                    BEQ     50334$
5231  015610                                      LET (R3)+ := #-1                ;INIT THE MSG PACKET WITH ALL 1'S
5232  015610  012723  177777                                                            MOV     #-1,(R3)
5233  015614                                      LET R2 := R2 + #2               ;UPDATE COUNTER.
5234  015614  062702  000002                                                            ADD     #2,R2
5235  015620                                    ENDDO
5236  015620  000770                                                                    BR      50333$
5237  015622                                                                    50334$:
5238  015622                                    LET @TSDB(R5) := #GSCPK           ;INITIATE A GET STATUS COMMAND.
5239  015622  012775  002320  002452                                                    MOV     #GSCPK,@
5240  015630  004737  011170                   JSR PC,WSSR                       ;WAIT A WHILE FOR SSR=1
5241  015634  004737  011224                   JSR PC,MOVMSG                     ;MOVE MSG PACKET TO COMMON AREA.
5242  015640                                    IF R4 EQ #X3.RNY THEN             ;IF WE HAVE A CAPSTAN RUNAWAY THEN:
5243  015640  020427  157400                                                            CMP     R4,#X3.R
5244  015644  001005                                                                    BNE     50335$
5245  015646                                      ERRDF #16.,RNYM,STAERM            ;REPORT CAPSTAN RUNAWAY WITH TACH CNT.
5246  015646  104455                                                                    TRAP    C$ERDF
5247  015650  000020                                                                    .WORD   16
5248  015652  004504                                                                    .WORD   RNYM
5249  015654  005372                                                                    .WORD   STAERM
5250  015656                                    ELSE                              ;ELSE IF NOT A RUNNAWAY:
5251  015656  000402                                                                    BR      50336$
5252  015660                                                                    50335$:
5253  015660  004737  015776                     JSR PC,PRXST                    ;PRINT EXTENDED STATUS REGISTERS.
5254  015664                                    ENDIF
5255  015664                                                                    50336$:
5256  015664                                    IFB RECLOG NE #0 THEN             ;IF THE RECORD HAS BEEN LOGGED THEN:
5257  015664  105737  003411                                                            TSTB    RECLOG
5258  015670  001404                                                                    BEQ     50337$
5259  015672                                      LET DROPED :B= DROPED + #1      ;SET UNIT DROPPED FLAG.
5260  015672  105237  003446                                                            INCB    DROPED
5261  015676  004737  014102                     JSR PC,LOG                      ;LOG DATA BYTES + RD WR ERRORS.
5262  015702                                    ENDIF
5263  015702                                                                    50337$:
5264  015702                                    DORPT                             ;PRINT PERFORMANCE REPORT
5265  015702  104424                                                                    TRAP    C$DRPT
```

```
5266  015704                          DROPUA: IF PASCNT(R5) NE #0 THEN
5267  015704  005765  003254                                                  TST     PASCNT(R
5268  015710  001402                                                          BEQ     50340$
5269  015712                            LET PASCNT(R5) := PASCNT(R5)  #1
5270  015712  005365  003254                                                  DEC     PASCNT(R
5271  015716                          ENDIF
5272  015716                                                              50340$:
5273  015716                          LET DROPN := DEVTBL(R5)           ;SAVE # OF UNIT TO BE DROPPED.
5274  015716  016537  002532  015774                                          MOV     DEVTBL(R
5275  015724                          LET R0 := R5 SHIFT  1             ;R0=LOGICAL DEVICE NUMBER
5276  015724  010500                                                          MOV     R5,R0
5277  015726  006200                                                          ASR     R0
5278  015730                          DODU R0                   ;DROP THE UNIT: EXEC BGNDU ENDDU CODE IF IDU = 0
5279  015730  104451                                                          TRAP    C$DODU
5280  015732                          IF DEVTBL(R5) NE #NINUSE THEN     ;IF UNIT NOT DROPPED
5281  015732  026527  002532  177774                                          CMP     DEVTBL(R
5282  015740  001410                                                          BEQ     50341$
5283  015742                            IFB IREC EQ #0 THEN               ;IF RECOVERY IS ENABLED THEN:
5284  015742  105737  002211                                                  TSTB    IREC
5285  015746  001005                                                          BNE     50342$
5286  015750  000240                             NOP
5287  015752  000240                             NOP
5288  015754  000240                             NOP
5289  015756                                     LET STAFLG :B= STAFLG . #1      ;SET START FLAG TO ENABLE REWIND.
5290  015756  105237  003452                                                  INCB    STAFLG
5291  015762                            ENDIF
5292  015762                                                              50342$:
5293  015762                          ENDIF
5294  015762                                                              50341$:
5295  015762                          DRORTN: LET DROPED :B= DROPED . #1       ;SET UNIT DROPPED FLAG.
5296  015762  105237  003446                                                  INCB    DROPED
5297  015766                                  LET R5 := R5SAVE            ;BTL
5298  015766  013705  003400                                                  MOV     R5SAVE,R
5299  015772  000207                         RTS     PC                  ;RETURN.
5300
5301  015774  000000                  DROPN: .WORD   0                   ;# OF UNIT TO BE DROPPED
```

```
5302                                    ;       SUBROUTINE TO PRINT EXTENDED STATUS REGISTERS.
5303                                    ;       INPUTS:
5304                                    ;       OUTPUTS:
5305                                    ;       REGISTERS:
5306                                    ;       CALLS:
5307
5308  015776                            PRXST:: PRINTX #GETSTM
5309  015776  012746  005157                                                  MOV     #GETSTM,
5310  016002  012746  000001                                                  MOV     #1,(SP)
5311  016006  010600                                                          MOV     SP,R0
5312  016010  104415                                                          TRAP    C$PNTX
5313  016012  062706  000004                                                  ADD     #4,SP
5314  016016                                  PRINTX  #STAER5,MSGPKT+MS.  ,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3
5315  016016  013746  002350                                                  MOV     MSGPKT+M
5316  016022  013746  002346                                                  MOV     MSGPKT+M
5317  016026  013746  002344                                                  MOV     MSGPKT+M
5318  016032  013746  002342                                                  MOV     MSGPKT+M
5319  016036  012746  006217                                                  MOV     #STAER5,
5320  016042  012746  000005                                                  MOV     #5,-(SP)
5321  016046  010600                                                          MOV     SP,R0
5322  016050  104415                                                          TRAP    C$PNTX
5323  016052  062706  000014                                                  ADD     #14,SP
5324  016056  000207                                  RTS PC
5325
5326
5327
5328
5329                                    ;       SUBROUTINE TO HALT AFTER EACH COMMAND.
5330                                    ;       INPUTS:
5331                                    ;       OUTPUTS:
5332                                    ;       REGISTERS:      R3, R4
5333                                    ;       CALLS:
5334
5335  016060                            CKHAE:: IFB HAE NE #0 THEN              ;IF HALT FLAG IS SET:
5336  016060  105737  002204                                                          TSTB    HAE
5337  016064  001430                                                                  BEQ     50343$
5338  016066                                            IFB MISCFG EQ #0 THEN         ;
5339  016066  105737  003455                                                          TSTB    MISCFG
5340  016072  001023                                                                  BNE     50344$
5341  016074                                              MANUAL                      ;IS MANUAL INTERVENTION ALLOWED?
5342  016074  104450                                                                  TRAP    C$MANI
5343  016076                                              BNCOMPLETE CKHRTN           ;BR IF NOT.
5344  016076  103023                                                                  BCC     CKHRTN
5345  016100                                              LET R4 := CMDWRD            ;COMMAND WORD.
5346  016100  013704  003346                                                          MOV     CMDWRD,R
5347  016104  004737  007416                                              JSR PC,GCMDA        ;FETCH ADR OF CMD ASCII.
5348  016110                                              LET HALTM :B= (R3)+         ;MOVE CMD ASCII
5349  016110  112337  004042                                                          MOVB    (R3)+,HA
5350  016114                                              LET HALTM+1 :B= (R3)+
5351  016114  112337  004043                                                          MOVB    (R3)+,HA
5352  016120                                              LET HALTM+2 :B= (R3)                ;INTO MESSAGE.
5353  016120  111337  004044                                                          MOVB    (R3),HAL
5354  016124                                              GMANIL HALT,TIME1,1,YES     ;HALT   WAIT FOR AN OEPRATOR INPUT.
5355  016124  104443                                                                  TRAP    C$GMAN
5356  016126  000404                                                                  BR      10000$
5357  016130  003364                                                                  .WORD   TIME1
```

```
5358 016132 000130                                                .WORD   T$CODE
5359 016134 00042                                                 .WORD   HALTM
5360 016136 000001                                                .WORD   1
5361 016140                          10000$:
5362 016140                                  ELSE                                    BR      5034$
5363 016140 000402                                                                   BR      5034$:
5364 016142
5365 016142                                  LET MISCFG :B= #0                       CLRB    MISCFG
5366 016142 105037 00345S
5367 016146                                  ENDIF
5368 016146                                  ENDIF                                   5034S:
5369 016146                                  ENDIF                                   5034S:
5370 0.6146                                                                          5034S:
5371 016146 000207          CKHRTN: RTS     PC          ;RETURN
5372                                 .EVEN
5373
5374 016150                          ENDMOD
```

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

016150                                          BGNMOD

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

                                                BGNRPT
016150                                  L$RPT::

                                                LET     R5SAVE := R5            ;SAVE CURRENT DEVICE POINTER.
016150  010537  003400                                                                  MOV     R5,R5SAV
016154  004737  015452                          JSR     PC,FIRSTU               ;FIND THE FIRST UNIT.
016160                                          WHILE DEVTBL(R5) NE #END DO     ;WHILE THERE ARE MORE DEVICES:
016160                                                                                  50346$:
016160  026527  002532  177777                                                          CMP     DEVTBL(R
016166  001562                                                                          BEQ     50347$
016170                                            PRINTS        #RPT1A,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
016170  016546  003324                                                                  MOV     RECCNT(R
016174  016546  003254                                                                  MOV     PASCNT(R
016200  016546  002532                                                                  MOV     DEVTBL(R
016204  012746  017012                                                                  MOV     #RPT1A,
016210  012746  000004                                                                  MOV     #4,-(SP)
016214  010600                                                                          MOV     SP,R0
016216  104416                                                                          TRAP    C$PNTS
016220  062706  000012                                                                  ADD     #12,SP
016224                                  PRINTS        #RPT1B,WRBC+30(R5),WRBC+20(R5),WRBC+10(R5),WRBC(R5)
016224  016546  002554                                                                  MOV     WRBC(R5)
016230  016546  002564                                                                  MOV     WRBC+10(
016234  016546  002574                                                                  MOV     WRBC+20(
016240  016546  002604                                                                  MOV     WRBC+30(
016244  012746  017067                                                                  MOV     #RPT1B,-
016250  012746  000005                                                                  MOV     #5, (SP)
016254  010600                                                                          MOV     SP,R0
016256  104416                                                                          TRAP    C$PNTS
016260  062706  000014                                                                  ADD     #14,SP
016264                                  PRINTS        #RPT1C,RRBC+30(R5),RRBC+20(R5),RRBC+10(R5),RRBC(R5)
016264  016546  002614                                                                  MOV     RRBC(R5)
016270  016546  002624                                                                  MOV     RRBC+10(
016274  016546  002634                                                                  MOV     RRBC+20(
016300  016546  002644                                                                  MOV     RRBC+30(
016304  012746  017140                                                                  MOV     #RPT1C,-
016310  012746  000005                                                                  MOV     #5,-(SP)
016314  010600                                                                          MOV     SP,R0
016316  104416                                                                          TRAP    C$PNTS
016320  062706  000014                                                                  ADD     #14,SP
016324                                  PRINTS        #RPT1D,RFBC+30(R5),RFBC+20(R5),RFBC+10(R5),RFBC(R5)
016324  016546  002654                                                                  MOV     RFBC(R5)
016330  016546  002664                                                                  MOV     RFBC+10(
016334  016546  002674                                                                  MOV     RFBC+20(
016340  016546  002704                                                                  MOV     RFBC+30(
```

```
5431  016344  012746  017211                                                                    MOV     #RPT1D,-
5432  016350  012746  000005                                                                    MOV     #5,-(SP)
5433  016354  010600                                                                           MOV     SP,R0
5434  016356  104416                                                                           TRAP    C$PNTS
5435  016360  062706  000014                                                                   ADD     #14,SP
5436  016364                    PRINTS        #RPT1F,WRREC(R5),RRREC(R5),RFREC(R5)
5437  016364  016546  002754                                                                   MOV     RFREC(R5
5438  016370  016546  002734                                                                   MOV     RRREC(R5
5439  016374  016546  002714                                                                   MOV     WRREC(R5
5440  016400  012746  017315                                                                   MOV     #RPT1F,
5441  016404  012746  000004                                                                   MOV     #4,-(SP)
5442  016410  010600                                                                           MOV     SP,R0
5443  016412  104416                                                                           TRAP    C$PNTS
5444  016414  062706  000012                                                                   ADD     #12,SP
5445  016420                    PRINTS        #RPT1G,WRUNR(R5),RRUNR(R5),RFUNR(R5)
5446  016420  016546  002764                                                                   MOV     RFUNR(R5
5447  016424  016546  002744                                                                   MOV     RRUNR(R5
5448  016430  016546  002724                                                                   MOV     WRUNR(R5
5449  016434  012746  017366                                                                   MOV     #RPT1G,
5450  016440  012746  000004                                                                   MOV     #4,-(SP)
5451  016444  010600                                                                           MOV     SP,R0
5452  016446  104416                                                                           TRAP    C$PNTS
5453  016450  062706  000012                                                                   ADD     #12,SP
5454  016454                    IFB BADTSW NE #0 THEN         ;
5455  016454  105737  002206                                                                   TSTB    BADTSW
5456  016460  001402                                                                           BEQ     50350$
5457  016462  004737  016544      JSR PC,BTRPT        ;GO PRINT BAD TAPE SPOTS WHEN  ENABLED
5458  016466                    ENDIF
5459  016466                                                                  50350$:
5460  016466                    PRINTS        #RPT1I,SCCNT(R5),HRDCNT(R5),FTLCNT(R5),VFYCNT(R5)
5461  016466  016546  003274                                                                   MOV     VFYCNT(R
5462  016472  016546  003314                                                                   MOV     FTLCNT(R
5463  016476  016546  003304                                                                   MOV     HRDCNT(R
5464  016502  016546  003264                                                                   MOV     SCCNT(R5
5465  016506  012746  017563                                                                   MOV     #RPT1I,-
5466  016512  012746  000005                                                                   MOV     #5,-(SP)
5467  016516  010600                                                                           MOV     SP,R0
5468  016520  104416                                                                           TRAP    C$PNTS
5469  016522  062706  000014                                                                   ADD     #14,SP
5470  016526  004737  015520      JSR   PC,NEXTU              ;FIND THE NEXT UNIT.
5471  016532                    ENDDO
5472  016532  000612                                                                           BR      50346$
5473  016534                                                                      50347$:
5474  016534                    LET R5 :  R5SAVE            ;RESTORE CURRENT DEVICE POINTER.
5475  016534  013705  003400                                                                   MOV     R5SAVE,R
5476  016540                    EXIT    RPT
5477  016540  000167                                                                           .WORD   J$JMP
5478  016542  001130                                                                           .WORD   L10010-2
5479
5480
5481
5482
5483                        ;       SUBR TO PRINT BAD TAPES SPOTS DURING THE REPORT PRINTS
5484                        ;       WRITE RETRIES: CUMULATIVE COUNT
5485                        ;       BAD TAPE SPOTS: COUNT PER TAPE PASS ONLY, NOT CUMULATIVE.
5486                        ;       COUNT OF RECOVERABLE WRITE ERRORS EXCLUDES BAD TAPE SPOTS.
```

```
5487
5488
5489  016544                                  BTRPT:  PRINTS  #RPT1E,WRTYCT(R5)       ;PRINT GLOBAL WRITE RETRY COUNT
5490  016544  016546  003244                                                                  MOV     WRTYCT(R
5491  016550  012746  017437                                                                  MOV     #RPT1E,
5492  016554  012746  000002                                                                  MOV     #2,(SP)
5493  016560  010600                                                                          MOV     SP,R0
5494  016562  104416                                                                          TRAP    C$PNTS
5495  016564  062706  000006                                                                  ADD     #6,SP
5496  016570                                          LET BTPT := BTADDR(R5)  ;BTPT IS BOTH THE BAD TAPE SPOT COUNTER
5497  016570  016537  002544  003436                                                          MOV     BTADDR(R
5498  016576                                          LET R3 := @BTPT SHIFT  1        ;AND THE LOGGING INDEX
5499  016576  017703  164634                                                                  MOV     @BTPT,R3
5500  016602  006203                                                                          ASR     R3
5501  016604                                          PRINTS  #RPT1J,R3       ;PRINT # OF BAD TAPE SPOTS
5502  016604  010346                                                                          MOV     R3,(SP)
5503  016606  012746  017467                                                                  MOV     #RPT1J,
5504  016612  012746  000002                                                                  MOV     #2,(SP)
5505  016616  010600                                                                          MOV     SP,R0
5506  016620  104416                                                                          TRAP    C$PNTS
5507  016622  062706  000006                                                                  ADD     #6,SP
5508  016626                                          IF R3 NE #0 THEN                ;PRINT RECORD # IF BAD SPOTS DETECTED
5509  016626  005703                                                                          TST     R3
5510  016630  001457                                                                          BEQ     50351$
5511  016632                                            IF R3 HI #20. THEN            ;
5512  016632  020327  000024                                                                  CMP     R3,#20.
5513  016636  101402                                                                          BLOS    50352$
5514  016640                                              LET R3 := #20.              ;20 BAD SPOTS IS THE LIMIT
5515  016640  012703  000024                                                                  MOV     #20.,R3
5516  016644                                            ENDIF
5517  016644                                                                                                50352$:
5518  016644                                            PRINTS        #CRLF$P         ;
5519  016644  012746  005216                                                                  MOV     #CRLF$P,
5520  016650  012746  000001                                                                  MOV     #1.(SP)
5521  016654  010600                                                                          MOV     SP,R0
5522  016656  104416                                                                          TRAP    C$PNTS
5523  016660  062706  000004                                                                  ADD     #4,SP
5524  016664                                          LET R4 := BTPT + #2     ;FETCH A BAD SPOT ID
5525  016664  013704  003436                                                                  MOV     BTPT,R4
5526  016670  062704  000002                                                                  ADD     #2,R4
5527  016674                                          LET R2 := #0            ;R2 = PRINT COUNT PER LINE: 10 MAX
5528  016674  005002                                                                          CLR     R2
5529  016676                                          REPEAT                          ;
5530  016676                                                                                        50353$:
5531  016676                                            PRINTS      #RPT1K,(R4)     ;PRINT A BAD SPOT ID
5532  016676  011446                                                                          MOV     (R4),(S
5533  016700  012746  017554                                                                  MOV     #RPT1K,-
5534  016704  012746  000002                                                                  MOV     #2,(SP)
5535  016710  010600                                                                          MOV     SP,R0
5536  016712  104416                                                                          TRAP    C$PNTS
5537  016714  062706  000006                                                                  ADD     #6,SP
5538  016720                                            LET R2 := R2 + #1     ;COUNT PRINTS
5539  016720  005202                                                                          INC     R2
5540  016722                                            LET R4 := R4 + #2     ;NEXT
5541  016722  062704  000002                                                                  ADD     #2,R4
5542  016726                                            IF R2 EQ #10. THEN    ;
```

```
5543  016726  020227  000012                                                          CMP     R2,#10.
5544  016732  001014                                                                  BNE     503548
5545  016734                          PRINTS #CRLFSP          ;GO TO NEXT PRINT LINE PAST 10 PRINTS
5546  016734  012746  005216                                                        MOV     #CRLFSP,
5547  016740  012746  000001                                                        MOV     #1.,(SP)
5548  016744  010600                                                              MOV     SP,R0
5549  016746  104416                                                              TRAP    C$PNTS
5550  016750  062706  000004                                                        ADD     #4,SP
5551  016754                          LET R3 := R3   #10.     ;ADJUST BAD SPOT COUNT
5552  016754  162703  000012                                                        SUB     #10.,R3
5553  016760                          LET R2 := R2   #10.     ;ADJUST PRINT COUNT
5554  016760  162702  000012                                                        SUB     #10.,R2
5555  016764                        ENDIF                     ;
5556  016764                                                              503548$:
5557  016764                      UNTIL R2 EQ R3              ;LIMIT: # OF BAD SPOTS
5558  016764  020203                                                              CMP     R2,R3
5559  016766  001343                                                              BNE     503538
5560  016770                    ENDIF                         ;
5561  016770                                                            503518$:
5562  016770                    PRINTS  #CRLF                 ;
5563  016770  012746  005213                                                      MOV     #CRLF,-(
5564  016774  012746  000001                                                      MOV     #1,-(SP)
5565  017000  010600                                                              MOV     SP,R0
5566  017002  104416                                                              TRAP    C$PNTS
5567  017004  062706  000004                                                      ADD     #4,SP
5568  017010  000207                    RTS PC
5569
5570
5571                                    .NLIST  BEX
      017012  047045  047045  040445  RPT1A:  .ASCIZ  /@N@N@AUNIT @D1@S3@APASS:@D5@S3@ARECORD:@D5@N/
      017067     045  041101  052131  RPT1B:  .ASCIZ  /@ABYTES WRITTEN  @D3@A,@Z3@A,@Z3@A,@Z3@N/
      017140  040445  054502  042524  RPT1C:  .ASCIZ  /@ABYTES READ REV @D3@A,@Z3@A,@Z3@A,@Z3@N/
      017211     045  041101  052131  RPT1D:  .ASCII  /@ABYTES READ FWD @D3@A,@Z3@A,@Z3@A,@Z3@N/
      017261     045  031123  022463          .ASCIZ  /@S23@AWRT@S4@ARDR@S4@ARDF@N/
      017315     045  051101  041505  RPT1F:  .ASCIZ  /@ARECOVERABLE ERRORS   @D5@S2@D5@S2@D5@N/
      017366  040445  047125  042522  RPT1G:  .ASCIZ  /@AUNRECOVERABLE ERRORS @D5@S2@D5@S2@D5@N/
      017437     045  053501  044522  RPT1E:  .ASCIZ  /@AWRITE RETRIES@S8@D5@N/
      017467     045  022516  031104  RPT1J:  .ASCIZ  /@N@D2@A BAD SPOTS THIS TAPE PASS PRECEDING RECORD #:/
      017554  042045  022465  030523  RPT1K:  .ASCIZ  /@D5@S1/
      017563     045  051501  042520  RPT1I:  .ASCII  "@ASPEC COND@S3@AHARD@S3@AFATAL@S3@ACOMPARE@N
      017637     045  031523  042045          .ASCIZ  /@S3@D5@S3@D5@S3@D5@S3@D5@N@N/
                                              .LIST   BEX
5572                                          .EVEN
5573
5574  017674                          ENDRPT
5575  017674                  L10010:
5576  017674  104425                                                              TRAP    C$RPT
5577
5578                            .SBTTL  LOAD DEVICE PROTECTION TABLE
5579
5580                            ;++
5581                            ;TABLE FOR SUPERVISOR TO IDENTIFY THE P TBL FOR THE LOAD DEV
5582                            ;THE SUPERVISOR USES THE TBL TO WARN THE OPERATOR WHEN HE TRIES TO TEST THE LOAD
5583                            ;
5584
5585  017676                            BGNPROT
```

```
5586  017676                          L$PROT::
5587  017676  000000                          .WORD 0          ;P TBL OFFSET OF TSSR, THE TS11 CSR
5588  017700  177777                          .WORD  1         ;P TBL OFFSET OF MASS BUS UNIT #:  1 = NOT A MAS
5589  017702  177777                          .WORD  1         ;P TBL OFFSET OF DRIVE #:  1 = NONE, ONE DRIVE P
5590  017704                          ENDPROT
```

```
5591                                            .SBTTL  INITIALIZE SECTION
5592
5593                                            ;**
5594                                            ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
5595                                            ; AT THE BEGINNING OF EACH PASS.
5596                                            ;
5597
5598  017704                                            BGNINIT
5599  017704                                    L$INIT::
5600
5601  017704                                    INIT10: IF @BIT0!BIT1 SETIN @CMDPKT THEN  ;IF CMD PACKET IS NOT ON MODULO 4 BOUN
5602  017704  032727  000003  002310                                                      BIT     @BIT0!BI
5603  017712  001426                                                                     BEQ     50355$
5604  017714                                              ERRSF #1,CMDPKM               ;PRINT ERROR MSG,
5605  017714  104454                                                                     TRAP    C$ERSF
5606  017716  000001                                                                     .WORD   1
5607  017720  004102                                                                     .WORD   CMDPKM
5608  017722  000000                                                                     .WORD   0
5609  017724  012746  000010                          MOV   #8., (SP)               ;SETUP STACK FOR LONG DELAY
5610  017730                                    97$:    DELAY 250.                    ;GO TO SUPERVISOR, WAIT 2 SECONDS.
5611  017730  012727  000372                                                          MOV     #250.,(P
5612  017734  000000                                                                  .WORD   0
5613  017736  013727  002116                                                          MOV     L$DLY,(P
5614  017742  000000                                                                  .WORD   0
5615  017744  005367  177772                                                          DEC     -6(PC)
5616  017750  001375                                                                  BNE     . 4
5617  017752  005367  177756                                                          DEC     -22(PC)
5618  017756  001367                                                                  BNE     .-20
5619  017760  005316                                    DEC   (SP)                    ;SUBTRACT 1
5620  017762  001362                                    BNE   97$                     ;BRANCH UNTIL DONE
5621  017764  005726                                    TST   (SP)+                   ;CLEAN UP THE STACK
5622  017766  000746                                    BR    INIT10                  ;
5623  017770                                    ENDIF
5624  017770                                                                                  50355$:
5625
5626  017770                                    IFB CLRFLG NE #0 THEN               ;IF CLR COUNTERS FLAG SET:
5627  017770  105737  002202                                                                 TSTB    CLRFLG
5628  017774  001413                                                                         BEQ     50356$
5629  017776  105037  002202                        CLRB  CLRFLG                    ;INIT CLR FLAG.
5630  020002                                        LET R2 := #0
5631  020002  005002                                                                         CLR     R2
5632  020004                                        WHILE R2 NE #CNTLEN DO
5633  020004                                                                                  50357$:
5634  020004  020227  000550                                                                 CMP     R2,#CNTL
5635  020010  001405                                                                         BEQ     50360$
5636  020012                                          LET WRBC(R2) := #0            ;CLR ALL STATISTICAL COUNTERS.
5637  020012  005062  002554                                                                 CLR     WRBC(R2)
5638  020016                                          LET R2 := R2 + #2
5639  020016  062702  000002                                                                 ADD     #2,R2
5640  020022                                        ENDDO
5641  020022  000770                                                                         BR      50357$
5642  020024                                    ENDIF                                 50360$:
5643  020024
5644  020024                                                                                  50356$:
5645
5646  020024                                    IFB RRANV NE #0 THEN                ;IF RESET RANDOM VARIABLE FLAG IS SET TH
```

```
5647  020024  105737  002203                                                                    TSTB    RRANV
5648  020030  001406                                                                            BEQ     50361$
5649  020032                                                                                    
5650  020032  012737  153624  003360                                                            MOV     @RANBC,R
5651  020040                                                                                    
5652  020040  012737  032561  00336?                                                            MOV     @RANSC,R
5653  020046                                                                                    
5654  020046                                                                                    
5655  020046                                                                                    
5656  020046  012700  000040                                                                    MOV     @EF.STAR
5657  020052  104447                                                                            TRAP    C$REFG
5658  020054                                                                                    
5659  020054  103026                                                                            BCC     INIT15
5660  020056                                                                                    
5661  020056  105237  003452                                                                    INCB    STAFLG
5662  020062                                                                                    
5663  020062  012705  000006                                                                    MOV     @6,R5
5664  020066                                                                                    
5665  020066                                                                                    
5666  020066                                                                                    
5667  020066  012765  177774  002532                                                            MOV     @NINUSE,
5668  020074                                                                                    
5669  020074  162705  000002                                                                    SUB     @2,R5
5670  020100                                                                                    
5671  020100  005705                                                                            TST     R5
5672  020102  001371                                                                            BNE     50362$
5673  020104                                                                                    
5674  020104  013705  002012                                                                    MOV     L$UNIT,R
5675  020110  006305                                                                            ASL     R5
5676  020112                                                                                    
5677  020112                                                                                    
5678  020112                                                                                    
5679  020112  162705  000002                                                                    SUB     @2,R5
5680  020116                                                                                    
5681  020116  010565  002532                                                                    MOV     R5,DEVTB
5682  020122  006265  002532                                                                    ASR     DEVTBL(R
5683  020126                                                                                    
5684  020126  005705                                                                            TST     R5
5685  020130  001370                                                                            BNE     50363$
5686                                                                                            
5687  020132                                                                                    
5688  020132  012700  000034                                                                    MOV     @EF.PWR,
5689  020136  104447                                                                            TRAP    C$REFG
5690  020140                                                                                    
5691  020140  103004                                                                            BCC     INIT16
5692  020142                                                                                    
5693  020142  105237  003452                                                                    INCB    STAFLG
5694  020146                                                                                    
5695  020146  105237  003453                                                                    INCB    PWRFLG
5696                                                                                            
5697  020152                                                                                    
5698  020152  104421                                                                            TRAP    C$RFLA
5699  020154  010037  003456                                                                    MOV     R0,OPFLA
5700  020160                                                                                    
5701  020160  005003                                                                            CLR     R3
5702  020162                                                                                    
```

```
                                                                                    ;RESET RANDOM BASE ).
  LET RANB := @RANBC
                                                                                    ;RESET RANDOM SAVE LOCATION.
  LET RANS := @RANSC
ENDIF
                                                                                50361$:
READEF  @EF.START                                                   ;READ START COMMAND EVENT FLAG.

BNCOMPLETE      INIT15                                              ;BRANCH IF NOT STARTING.

LET STAFLG :B= STAFLG + @1                                          ;SET START COMMAND FLAG.

LET R5 := @6
                                                                    ;INITIATE UNIT NUMBER TABLE
REPEAT                                                                  50362$:
                                                                    ;BY STORING NOT IN USE IN EACH LOCATION.
  LET DEVTBL(R5) := @NINUSE
  LET R5 := R5 - @2

UNTIL R5 EQ @0

                                                                    ;STORE ALL UNIT
LET R5 := L$UNIT SHIFT 1                                                50363$:
                                                                    ;NUMBERS IN DEVTBL.
REPEAT
  LET R5 := R5 - @2
    LET DEVTBL(R5) := R5 SHIFT  1

  UNTIL R5 EQ @0

INIT15: READEF @EF.PWR                                              ;HAS THERE BE A POWER FAILURE?

  BNCOMPLETE INIT16                                                 ;BRANCH IF NOT.

    LET STAFLG :B= STAFLG + @1                                        ;IF SO, SET THE START FLAG.

    LET PWRFLG :B= PWRFLG + @1                                      ;IF SO - SET THE POWER FAIL FLAG.

INIT16: RFLAGS  OPFLAG                                              ;READ AND STORE FLAGS SET BY OPERATOR

  LET R3 := @0                                                      ;CLEAR EVENT FLAG

    IFB PWRFLG EQ @0 THEN                                           ;IF POWER FAIL HAS NOT OCCURRED THEN:
```

```
5703  020162  105737  003453                                                  TSTB    PWRFLG
5704  020166  001020                                                          BNE     50364$
5705  020170
5706  020170  012700  000035    READEF #EF.NEW          ;UPDATE PASS COUNT WHEN   MOV     #EF.NEW,
5707  020174  104447                                                          TRAP    C$REFG
5708  020176
5709  020176  103014    IFCOND CS THEN          ;SUPERVISOR IS IN NEW PASS      BCC     50365$
5710  020200
5711  020200  105737  003452    IFB STAFLG EQ #0 THEN   ;AND DIAG WAS NEITHER STARTED
                                                                              TSTB    STAFLG
5712  020204  001010                                                          BNE     50366$
5713  020206
5714  020206  012700  000037      READEF #EF.RES        ;NOR                    MOV     #EF.RES,
5715  020212  104447                                                          TRAP    C$REFG
5716  020214
5717  020214  103402      IFCOND CC THEN        ;RESTARTED                    BCS     50367$
5718  020216
5719  020216  005103        LET R3 := COMP R3   ;DO IT                        COM     R3
5720  020220
5721  020220  000401      ELSE                                                BR      50370$
5722  020222                                                        50367$:
5723  020222
5724  020222  005203        LET R3 := R3 + #1   ;SET 1ST PASS IF NEW PASS AND  INC     R3
5725  020224
5726  020224      ENDIF                       ;RESTARTING
                                                              50370$:
5727  020224
5728  020224  000401    ELSE                                                  BR      50371$
5729  020226                                                        50366$:
5730  020226
5731  020226  005203      LET R3 := R3 + #1   ;SET 1ST PASS IF NEW PASS AND    INC     R3
5732  020230
5733  020230    ENDIF                         ;STARTING
                                                              50371$:
5734  020230
5735  020230  ENDIF                          ;DO NOT UPDATE IT ON CONTINUE
                                                              50365$:
5736  020230
5737  020230                                 ;OR ON POWER FAIL
                                                              50364$:
5738  020230  004737  015452  JSR     PC,FIRSTU               ;INIT DEVICE POINTER.
5739  020234                  LET R2 := #0                    ;INIT DEVICE COUNTER.
5740  020234  005002                                                          CLR     R2
5741  020236
5742  020236                  WHILE DEVTBL(R5) NE #END DO
                                                              50372$:
5743  020236  026527  002532  177777                                        CMP     DEVTBL(R
5744  020244  001450                                                          BEQ     50373$
5745  020246
5746  020246  005202    LET R2 := R2 + #1                                     INC     R2
5747  020250
5748  020250  010500    LET R0 := R5 SHIFT  1                                 MOV     R5,R0
5749  020252  006200                                                          ASR     R0
5750  020254
5751  020254  104442    GPHARD  R0,R0                   ;GET HARDWARE P TABLE FROM SUPER.
                                                                              TRAP    C$GPHRD
5752  020256
5753  020256  103036    IFCOND CS THEN                                        BCC     50374$
5754  020260
5755  020260  011065  002462      LET TSSR(R5) := (R0)       ;SAVE TSSR ADDRESS.
                                                                              MOV     (R0),TSS
5756  020264
5757  020264  012065  002452      LET TSDB(R5) := (R0)+   #2  ;SAVE TSDB ADDRESS.
                                                                              MOV     (R0)+,TS
5758  020270  162765  000002  002452                                         SUB     #2,TSDB(
```

```
5759  020276
5760  020276  011065  002472                    LET TSVCT(R5) := (R0)        ;SAVE INTERRUPT VECTOR ADDRESS.
5761  020302                                                                    MOV     (R0),TSV
5762  020302  012746  000340                    SETVEC TSVCT(R5),TS4INT(R5),#INTPRI ;SET UP INTERUPT PROCESSING COND
5763  020306  016546  002512                                                    MOV     #INTPRI,
5764  020312  016546  002472                                                    MOV     TS4INT(R
5765  020316  012746  000003                                                    MOV     TSVCT(R5
5766  020322  104437                                                           MOV     #3,-(SP)
5767  020324  062706  000010                                                    TRAP    C$SVEC
5768  020330                                    IF R3 NE #0 THEN             ;ACTUAL PASSCOUNT UPDATE PER R3
5769  020330  005703                                                           ADD     #10,SP
5770  020332  001410                                                           TST     R3
5771  020334                                      IF R3 LT #0 THEN             BEQ     50375$
5772  020334  005703                                                           TST     R3
5773  020336  002003                                                           BGE     50376$
5774  020340                                        LET PASCNT(R5) := PASCNT(R5) - #1
5775  020340  005265  003254                                                   INC     PASCNT(R
5776  020344                                      ELSE
5777  020344  000403                                                           BR      50377$
5778  020346                                                          50376$:
5779  020346                                        LET PASCNT(R5) := #1
5780  020346  012765  000001  003254                                           MOV     #1,PASCN
5781  020354                                      ENDIF
5782  020354                                                          50377$:
5783  020354                                    ENDIF
5784  020354                                                          50375$:
5785  020354                                  ENDIF
5786  020354                                                          50374$:
5787  020354                                  LET RECCNT(R5) := #0          ;CLEAR RECORD COUNT
5788  020354  005065  003324                                                   CLR     RECCNT(R
5789  020360  004737  015520                  JSR   PC,NEXTU               ;DO IT FOR ALL DEVICES.
5790  020364                                  ENDDO
5791  020364  000724                                                           BR      50372$
5792  020366                                                          50373$:
5793
5794  020366                                  IF R2 EQ #0 THEN             ;IF THERE ARE NO UNITS:
5795  020366  005702                                                           TST     R2
5796  020370  001033                                                           BNE     50400$
5797  020372                                    PRINTF #AUDRPM             ;PRINT ALL UNITS DROPPED.
5798  020372  012746  004645                                                   MOV     #AUDRPM,
5799  020376  012746  000001                                                   MOV     #1,-(SP)
5800  020402  010600                                                           MOV     SP,R0
5801  020404  104417                                                           TRAP    C$PNTF
5802  020406  062706  000004                                                   ADD     #4,SP
5803  020412  012746  000010                    MOV   #8.,-(SP)            ;SETUP STACK FOR LONG DELAY
5804  020416                            98$:      DELAY 250.                 ;GO TO SUPERVISOR, WAIT 2 SECONDS.
5805  020416  012727  000372                                                   MOV     #250.,(P
5806  020422  000000                                                           .WORD   0
5807  020424  013727  002116                                                   MOV     L$DLY,(P
5808  020430  000000                                                           .WORD   0
5809  020432  005367  177772                                                   DEC     6(PC)
5810  020436  001375                                                           BNE     .-4
5811  020440  005367  177756                                                   DEC     -22(PC)
5812  020444  001367                                                           BNE     .-20
5813  020446  005316                    DEC   (SP)                 ;SUBTRACT 1
5814  020450  001362                    BNE   98$                  ;BRANCH UNTIL DONE
```

```
5815  020452  005726                          TST   (SP)+                     ;CLEAN UP THE STACK
5816  020454                                  BREAK                           ;GO TO SUPERVISOR, CHECK TTY.
5817  020454  104422                                                                  TRAP    C$BRK
5818  020456                                DOCLN                             ;DO CLEAN CODE + ABORT PASS.
5819  020456  104444                                                                  TRAP    C$DCLN
5820  020460                                ENDIF
5821  020460                                                                  50400$:
5822
5823  020460                                SETPRI #PRI00                     ;LOWER CPU PRIORITY TO 0
5824  020460  012700  000000                                                         MOV     #PRI00,R
5825  020464  104441                                                                 TRAP    C$SPRI
5826  020466                                IFB IREC EQ #0 AND #ADR NOTSETIN OPFLAG THEN    ;IF ERROR RECOVERY IS EN
5827  020466  105737  002211                                                       TSTB    IREC
5828  020472  001152                                                               BNE     50401$
5829  020474  032737  000020  003456                                               BIT     #ADR,OPF
5830  020502  001146                                                               BNE     50401$
5831  020504  004737  015452                  JSR PC,FIRSTU                     ;AND AUTO DROP NOT CALLED, THEN SET UP F
5832  020510                                  WHILE DEVTBL(R5) NE #END DO       ;WHILE THERE ARE MORE DEVICES:
5833  020510                                                                  50402$:
5834  020510  026527  002532  177777                                               CMP     DEVTBL(R
5835  020516  001540                                                               BEQ     50403$
5836  020520                                    BEGIN COUNTER                   ;START 3.5 MINUTE COUNTER
5837  020520                                      INCR TIME1 FROM #1 TO #25 BY #1
5838  020520  012737  000001  003364                                               MOV     #1,TIME1
5839  020526  000402                                                               BR      50405$
5840  020530                                                                  50406$:
5841  020530  005237  003364                                                       INC     TIME1
5842  020534                                                                  50405$:
5843  020534  023727  003364  000025                                               CMP     TIME1,#2
5844  020542  003113                                                               BGT     50407$
5845  020544                                      LET @TSDB(R5) := #GSCPK       ;AND GET UNITS STATUS
5846  020544  012775  002320  002452                                               MOV     #GSCPK,@
5847  020552                                    DELAY 1                         ;WAIT
5848  020552  012727  000001                                                       MOV     #1,(PC)+
5849  020556  000000                                                               .WORD   0
5850  020560  013727  002116                                                       MOV     L$DLY,(P
5851  020564  000000                                                               .WORD   0
5852  020566  005367  177772                                                       DEC     -6(PC)
5853  020572  001375                                                               BNE     .-4
5854  020574  005367  177756                                                       DEC     -22(PC)
5855  020600  001367                                                               BNE     .-20
5856  020602                                    IF #TS.SSR SETIN @TSSR(R5) THEN
5857  020602  032775  000200  002462                                               BIT     #TS.SSR,
5858  020610  001420                                                               BEQ     50410$
5859  020612                                      IF #TS.OFL NOTSETIN @TSSR(R5) THEN
5860  020612  032775  000100  002462                                               BIT     #TS.OFL,
5861  020620  001001                                                               BNE     50411$
5862  020622                                        LEAVE COUNTER                 ;EXIT COUNTER WHEN UNIT ON LINE
5863  020622  000463                                                               BR      50404$
5864  020624                                      ELSE
5865  020624                                                                  50411$:
5866  020624                                        PRINTF #OFLINM,DEVTBL(R5)   ;PRINT UNIT OFF LINE EVERY 10 SEC
5867  020624  016546  002532                                                       MOV     DEVTBL(R
5868  020630  012746  005127                                                       MOV     #OFLINM,
5869  020634  012746  000002                                                       MOV     #2,-(SP)
586C  020640  010600                                                               MOV     SP,R0
```

```
5871  020642  104417                                                          TRAP    C$PNTF
5872  020644  062706  000006                                                  ADD     #6,SP
5873  020650                                  ENDIF
5874  020650                                                          50412$:
5875  020650                              ELSE
5876  020650  000412                                                          BR      50413$
5877  020652                                                          50410$:
5878  020652                                  PRINTF #NRDYM,DEVTBL(R5)
5879  020652  016546  002532                                                  MOV     DEVTBL(R
5880  020656  012746  021616                                                  MOV     #NRDYM,-
5881  020662  012746  000002                                                  MOV     #2,-(SP)
5882  020666  010600                                                          MOV     SP,R0
5883  020670  104417                                                          TRAP    C$PNTF
5884  020672  062706  000006                                                  ADD     #6,SP
5885  020676                              ENDIF
5886  020676                                                          50413$:
5887  020676                              INCR TIME2 FROM #1 TO #13 BY #1
5888  020676  012737  000001  003366                                          MOV     #1,TIME2
5889  020704  000402                                                          BR      50414$
5890  020706                                                          50415$:
5891  020706  005237  003366                                                  INC     TIME2
5892  020712                                                          50414$:
5893  020712  023727  003366  000013                                          CMP     TIME2,#1
5894  020720  003023                                                          BGT     50416$
5895  020722  012746  000004                  MOV   #4.,(SP)          ;SETUP STACK FOR LONG DELAY
5896  020726                          99$:    DELAY 250.              ;GO TO SUPERVISOR, WAIT 1 SECOND
5897  020726  012727  000372                                                  MOV     #250.,(P
5898  020732  000000                                                          .WORD   0
5899  020734  013727  002116                                                  MOV     L$DLY,(P
5900  020740  000000                                                          .WORD   0
5901  020742  005367  177772                                                  DEC     -6(PC)
5902  020746  001375                                                          BNE     .-4
5903  020750  005367  177756                                                  DEC     -22(PC)
5904  020754  001367                                                          BNE     .-20
5905  020756  005316                      DEC   (SP)              ;SUBTRACT 1
5906  020760  001362                      BNE   99$               ;BRANCH UNTIL DONE
5907  020762  005726                      TST   (SP)+             ;CLEAN UP THE STACK
5908  020764                              BREAK           ;ALLOW TERMINAL INTERRUPT
5909  020764  104422                                                          TRAP    C$BRK
5910  020766                          .   ENDINC
5911  020766  000747                                                          BR      50415$
5912  020770                                                          50416$:
5913  020770                              ENDINC
5914  020770  000657                                                          BR      50406$
5915  020772                                                          50407$:
5916  020772                          END COUNTER
5917  020772                                                          50404$:
5918  020772                          IF TIME1 GT #25 THEN        ;IF OFF LINE FOR 3.5 MINUTES
5919  020772  023727  003364  000025                                          CMP     TIME1,#2
5920  021000  003404                                                          BLE     5041?$
5921  021002  004737  011224              JSR PC,MOVMSG           ;GET MESSAGE PACKET
5922  021006  004737  011736              JSR PC,TCC1             ;PRINT ERROR AND DROP OFF LINE UNIT
5923  021012                          ENDIF
5924  021012                                                          5041?$:
5925                                      ;REPEAT UNTIL ON LINE OR TIMED OUT.
5926  021012  004737  015520          JSR PC,NEXTU                ;SET UP FOR NEXT UNIT.
```

```
5927  021016                                                    ENDDO
5928  021016  000634                                                      BR      50402$
5929  021020                                                  50403$:
5930  021020                                                ENDIF
5931  021020                                                  50401$:
5932  021020                                                IFB PWRFLG EQ #0 THEN
5933  021020  105737  003453                                              TSTB    PWRFLG
5934  021024  001026                                                      BNE     50420$
5935  021026                                                  MEMORY DATAWT                   ;REQUEST MEMORY FROM SUPER FOR RD/WR BUF
5936  021026  104431                                                      TRAP    C$MEM
5937  021030  010037  003334                                              MOV     R0,DATAW
5938  021034                                                  LET DATARD :- DATAWT + #DATCNT          ;SET RD BFR AD
5939  021034  013737  003334  003336                                      MOV     DATAWT,D
5940  021042  062737  004000  003336                                      ADD     #DATCNT,
5941  021050                                                  IF @DATAWT LT #DATCNT THEN      ;WHEN NOT ENOUGH FREE MEMO AVAILABLE
5942  021050  027727  162260  004000                                      CMP     @DATAWT,
5943  021056  002011                                                      BGE     50421$
5944  021060                                                    PRINTF #MEMOM                 ;WARN OPERATOR
5945  021060  012746  021126                                              MOV     #MEMOM,
5946  021064  012746  000001                                              MOV     #1,-(SP)
5947  021070  010600                                                      MOV     SP,R0
5948  021072  104417                                                      TRAP    C$PNTF
5949  021074  062706  000004                                              ADD     #4,SP
5950  021100                                                    DOCLN                         ;AND ABORT PASS
5951  021100  104444                                                      TRAP    C$DCLN
5952  021102                                                  ENDIF         ;DIAG MUST BE RE LOADED IN A CPU WITH LARGER MEMO
5953  021102                                                  50421$:
5954  021102                                                ENDIF
5955  021102                                                  50420$:
5956
5957  021102                                                LET CHGFLG :B= #0                ;CLR CHANGE CMD SEQ TBL FLAG.
5958  021102  105037  002212                                              CLRB    CHGFLG
5959  021106                                                  LET R3 := #ENDFLG
5960  021106  012703  003452                                              MOV     #ENDFLG,
5961  021112  004737  011154                                  JSR     PC,CLRERR               ;CLEAR ALL FLAGS.
5962  021116                                                  LET PWRFLG :B= #0               ;CLEAR THE POWER FAIL FLAG.
5963  021116  105037  003453                                              CLRB    PWRFLG
5964
5965  021122                                                  EXIT    INIT
5966  021122  104432                                                      TRAP    C$EXIT
5967  021124  000104                                                      .WORD   L10012 .
```

```
 5968
 5969  021126  040445  051106  042505          MEMOM:  .ASCII  /%AFREE MEMO TOO SMALL FOR RD WR BFRS%N/
 5970  021134  046440  046505  020117
 5971  021142  047524  020117  046523
 5972  021150  046101  020114  047506
 5973  021156  020122  042122  053455
 5974  021164  020122  043102  051522
 5975  021172  047045
 5976  021174  040445  042522  046055                  .ASCIZ  /%ARE-LOAD IN LARGER MEMO%N/
 5977  021202  040517  020104  047111
 5978  021210  046040  051101  042507
 5979  021216  020122  042515  047515
 5980  021224  047045     000
 5981          021230                                  .EVEN
 5982
 5983  021230                                          ENDINIT
 5984  021230                                  L10012:
 5985  021230  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
                                        ;
021232                                          BGNAUTO
021232                                  L$AUTO::
                                                JSR PC,FIRSTU                   ;FIND FIRST UNIT
                                                WHILE DEVTBL(R5) NE #END DO     ;
                                                                                        50422$:
021236  026527  002532  177777                                                            CMP     DEVTBL(R
021244  001525                                                                           BEQ     50423$
                                                  LET TRAPD4 :B= #0             ;
021246  105037  003454                                                                   CLRB    TRAPD4
                                                  SETVEC #4,#TRAP4,#PRIO7       ;SET VECTOR 4
021252  012746  000340                                                                   MOV     #PRIO7,-
021256  012746  021646                                                                   MOV     #TRAP4,-
021262  012746  000004                                                                   MOV     #4,-(SP)
021266  012746  000003                                                                   MOV     #3,-(SP)
021272  104437                                                                          TRAP    C$SVEC
021274  062706  000010                                                                   ADD     #10,SP
                                                  LET R2 := @TSSR(R5)           ;ADDRESS TS11 INTERFACE
021300  017502  002462                                                                   MOV     @TSSR(R5
                                                  CLRVEC #4                     ;CLEAR VECTOR AT 4
021304  012700  000004                                                                   MOV     #4,R0
021310  104436                                                                          TRAP    C$CVEC
021312                                            IFB TRAPD4 NE #0 THEN
021312  105737  003454                                                                   TSTB    TRAPD4
021316  001423                                                                          BEQ     50424$
021320                                            LET FTLCNT(R5) := FTLCNT(R5) + #1
021320  005265  003314                                                                   INC     FTLCNT(R
021324                                            PRINTF #AUTODM,TSSR(R5)     ;PRINT ERROR
021324  016546  002462                                                                   MOV     TSSR(R5)
021330  012746  021522                                                                   MOV     #AUTODM,
021334  012746  000002                                                                   MOV     #2,-(SP)
021340  010600                                                                          MOV     SP,R0
021342  104417                                                                          TRAP    C$PNTF
021344  062706  000006                                                                   ADD     #6,SP
021350                                            LET DROPN :- DEVTBL(R5)             ;SAVE # OF UNIT TO BE DROPPED.
021350  016537  002532  015774                                                            MOV     DEVTBL(R
021356                                            LET R0 := R5 SHIFT  1               ;R0-LOGICAL DEVICE NUMBER
021356  010500                                                                          MOV     R5,R0
021360  006200                                                                          ASR     R0
021362                                            DODU R0                     ;DROP THE UNIT: EXEC BGNDU-ENDDU CODE IF
021362  104451                                                                          TRAP    C$DODU
021364                                          ELSE
021364  000452                                                                          BR      50425$
021366                                                                          50424$:
021366                                            LET @TSDB(R5) :- #GSCPK       ;SEND GET STATUS COMMAND
021366  012775  002320  002452                                                          MOV     #GSCPK,@
021374  004737  011170                          JSR PC,WSSR                   ;WAIT
021400                                          IF #TS.SSR SETIN @TSSR(R5) THEN
```

```
 6042  021400  032775  000200  002462                                                    BIT     #TS.SSR,
 6043  021406  001423                                                                    BEQ     50426$
 6044  021410                                  IF #TS.OFL SETIN @TSSR(R5) THEN
 6045  021410  032775  000100  002462                                                    BIT     #TS.OFL,
 6046  021416  001416                                                                    BEQ     50427$
 6047  021420                                    LET FTLCNT(R5) := FTLCNT(R5) + #1
 6048  021420  005265  003314                                                          INC     FTLCNT(R
 6049  021424                                    PRINTF #OFLINM,DEVTBL(R5)
 6050  021424  016546  002532                                                          MOV     DEVTBL(R
 6051  021430  012746  005127                                                          MOV     #OFLINM,
 6052  021434  012746  000002                                                          MOV     #2,-(SP)
 6053  021440  010600                                                                  MOV     SP,R0
 6054  021442  104417                                                                  TRAP    C$PNTF
 6055  021444  062706  000006                                                          ADD     #6,SP
 6056  021450  004737  015704                                    JSR PC,DROPUA
 6057  021454                                  ENDIF
 6058  021454                                                                  50427$:
 6059  021454                                ELSE
 6060  021454  000416                                                                  BR      50430$
 6061  021456                                                                  50426$:
 6062  021456                                    LET FTLCNT(R5) := FTLCNT(R5) + #1
 6063  021456  005265  003314                                                          INC     FTLCNT(R
 6064  021462                                    PRINTF #NRDYM,DEVTBL(R5)
 6065  021462  016546  002532                                                          MOV     DEVTBL(R
 6066  021466  012746  021616                                                          MOV     #NRDYM,-
 6067  021472  012746  000002                                                          MOV     #2,-(SP)
 6068  021476  010600                                                                  MOV     SP,R0
 6069  021500  104417                                                                  TRAP    C$PNTF
 6070  021502  062706  000006                                                          ADD     #6,SP
 6071  021506  004737  015704                                    JSR PC,DROPUA
 6072  021512                                ENDIF
 6073  021512                                                                  50430$:
 6074  021512                                  ENDIF
 6075  021512                                                                  50425$:
 6076  021512  004737  015520                                    JSR PC,NEXTU
 6077  021516                                ENDDO
 6078  021516  000647                                                                  BR      50422$
 6079  021520                                                                  50423$:
 6080
 6081  021520                                ENDAUTO
 6082  021520                            L10013:
 6083  021520  104461                                                                  TRAP    C$AUTO
 6084
 6085  021522  040445  052502  020123      AUTODM: .ASCII  /%ABUS TRAP AT %O6%N/
 6086  021530  051124  050101  040440
 6087  021536  020124  047445  022466
 6088  021544     116
 6089  021545     045  044501  052116              .ASCIZ  /%AINTERFACE BAD OR NOT SET TO ABOVE AD%N/
 6090  021552  051105  040506  042503
 6091  021560  041040  042101  047440
 6092  021566  020122  047516  020124
 6093  021574  042523  020124  047524
 6094  021602  040440  047502  042526
 6095  021610  040440  022504  000116
 6096  021616  040445  047125  052111      NRDYM:  .ASCIZ  /%AUNIT %D1%A NOT RDY%N/
 6097  021624  022440  030504  040445
```

```
6098  021632  047040  052117  051040
6099  021640  054504  047045     000
6100          021646                        .EVEN
6101
6102                                  ;       DEVICE BUS TRAP HANDLER
6103                                  ;       OUTPUT: TRAPD4 BYTE  1: TRAPED AT 4
6104                                  ;                            0: NO TRAP
6105
6106  021646                          TRAP4:: LET TRAPD4 :B= TRAPD4 + #1
6107  021646  105237  003454                                                  INCB    TRAPD4
6108  021652  000002                        RTI
6109
6110
6111
```

```
                                        .SBTTL  CLEANUP CODING SECTION

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

021654                                          BGNCLN
021654                                  L$CLEAN::

021654  004737  015452                          JSR     PC,FIRSTU               ;FIND FIRST UNIT.
021660                                          WHILE DEVTBL(R5) NE #END DO
021660                                                                                  50431$:
021660  026527  002532  177777                                                            CMP     DEVTBL(R
021666  001410                                                                          BEQ     50432$
021670  004737  011170                            JSR PC,WSSR                     ;WAIT FOR UNIT READY OR TIMEOUT.
021674                                            CLRVEC      TSVCT(R5)           ;RELEASE INTERRUPT VECTORS FOR ALL DEV.
021674  016500  002472                                                            MOV     TSVCT(R5
021700  104436                                                                    .RAP    C$CVEC
021702  004737  015520                            JSR   PC,NEXTU                  ;FIND NEXT UNIT.
021706                                          ENDDO
021706  000764                                                                    BR      50431$
021710                                                                                  50432$:

021710                                          EXIT    CLN
021710  104432                                                                    TRAP    C$EXIT
021712  000002                                                                    .WORD   L10014 .

021714                                          .EVEN

021714                                          ENDCLN
021714                                  L10014:
021714  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
                                        ;

021716                                          BGNDU
021716                                  L$DU::

021716                                          LET R5 := R0 SHIFT 1            ;R5 = LOGICAL DEVICE NUMBER X 2.
021716  010005                                                                          MOV     R0,R5
021720  006305                                                                          ASL     R5
021722                                          LET DEVTBL(R5) := #NINUSE       ;SET NOT IN USE FLAG FOR THE DEVICE.
021722  012765  177774  002532                                                        MOV     #NINUSE,
021730                                          CLRVEC TSVCT(R5)                ;RELEASE THE INTERRUPT VECTOR.
021730  016500  002472                                                                MOV     TSVCT(R5
021734  104436                                                                      TRAP    C$CVEC
021736                                          PRINTF  #DROPDM,DROPN           ;PRINT DROP DEVICE MESSAGE
021736  013746  015774                                                                MOV     DROPN,-(
021742  012746  004616                                                                MOV     #DROPDM,
021746  012746  000002                                                                MOV     #2, (SP)
021752  010600                                                                      MOV     SP,R0
021754  104417                                                                      TRAP    C$PNTF
021756  062706  000006                                                             ADD     #6,SP

021762                                          EXIT    DU
021762  000167                                                                      .WORD   J$JMP
021764  000000                                                                      .WORD   L10015-2

                                        .EVEN

021766                                          ENDDU
021766                                  L10015:
021766  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.
                                        ;-

021770                                          BGNAU
021770                                  L$AU::

                                                LET R5 := R0 SHIFT 1            ;R5 = LOGICAL DEVICE NUMBER X 2.
021770  010005                                                                          MOV     R0,R5
021772  006305                                                                          ASL     R5
021774                                          LET DEVTBL(R5) := R0    ;STORE UNIT # IN DEVICE TABLE.
021774  010065  002532                                                                  MOV     R0,DEVTB
022000                                          GPHARD  R0,R0                   ;GET HARDWARE P TABLE FROM SUPER.
022000  104442                                                                          TRAP    C$GPHRD
022002                                          LET TSSR(R5) := (R0)            ;SAVE TSSR ADDRESS.
022002  011065  002462                                                                  MOV     (R0),TSS
022006                                          LET TSDB(R5) := (R0)+ - #2      ;SAVE TSDB ADDRESS.
022006  012065  002452                                                                  MOV     (R0)+,TS
022012  162765  000002  002452                                                          SUB     #2,TSDB(
022020                                          LET TSVCT(R5) := (R0)           ;SAVE INTERRUPT VECTOR ADDRESS.
022020  011065  002472                                                                  MOV     (R0),TSV
022024                                          SETVEC TSVCT(R5),TS4INT(R5),#INTPRI     ;SET UP INTERUPT PROCESSING COND
022024  012746  000340                                                                  MOV     #INTPRI,
022030  016546  002512                                                                  MOV     TS4INT(R
022034  016546  002472                                                                  MOV     TSVCT(R5
022040  012746  000003                                                                  MOV     #3,-(SP)
022044  104437                                                                          TRAP    C$SVEC
022046  062706  000010                                                                  ADD     #10,SP
022052                                          LET INTFLG(R5) := #0            ;CLEAR INTERRUPT FLAGS.
022052  005065  003416                                                                  CLR     INTFLG(R

022056                                          EXIT    AU
022056  000167                                                                          .WORD   J$JMP
022060  000000                                                                          .WORD   L10016-2

                                                .EVEN

022062                                          ENDAU
022062                                  L10016:
022062  104452                                                                          TRAP    C$AU

022064                                          ENDMOD
```

```
6228
6229                                    .TITLE HARDWARE TESTS
6230
6231                                    .SBTTL  TEST 1:  BASIC FUNCTIONS.
6232
6233                                    ;++
6234                                    ; TEST TO EXECUTE ALL TS04 FUNCTIONS.
6235                                    ;-
6236
6237  022064                                    BGNMOD
6238
6239  022064                                    BGNTST
6240  022064                            T1::
6241
6242  022064                                    LET RANDOM :B= #0                 ;CLR THE RANDOM OPERATIONS FLAG.
6243  022064  105037  003441                                                             CLRB    RANDOM
6244  022070                                    LET EXPBOT :B= #0                 ;CLR EXPECT BOT FLAG.
6245  022070  105037  003440                                                             CLRB    EXPBOT
6246
6247  022074                                    BGNSUB                            ;SUBTEST 1 - SET CHAR, DRIVE INIT, GET S
6248  022074                            T1.1:
6249  022074  104402                                                             TRAP    C$BSUB
6250
6251  022076                                    LET R2 := #BFSEQ0                 ;ADR OF CMD SEQ.
6252  022076  012702  022722                                                             MOV     #BFSEQ0,
6253  022102  004737  022676                    JSR     PC,BFSEQ                  ;SET UP CMD SEQ.
6254  022106  004737  006510                    JSR     PC,EXALL                  ;EXECUTE CMD SEQ ON ALL DEVICES.
6255  022112  004737  015452                    JSR     PC,FIRSTU                 ;FIND THE FIRST UNIT.
6256  022116                                    WHILE DEVTBL(R5) NE #END DO       ;WHILE THERE ARE MORE DEVICES:
6257  022116                                                                                50433$:
6258  022116  026527  002532  177777                                                       CMP     DEVTBL(R
6259  022124  001434                                                                       BEQ     50434$
6260  022126
6261  022126  016502  002502                        LET R2 := MSGPKA(R5)          ;GET MSG PACKET ADR.
6262  022132                                                                             MOV     MSGPKA(R
6263  022132  062702  000012                        LET R2 := R2 + #12            ;GET XSTAT2 ADR.
6264  022136                                                                             ADD     #12,R2
6265  022136  011265  002522                        LET TS4CL(R5) := (R2) CLR.BY #177400  ;STORE CODE LEVEL FROM DTR BYTE.
6266  022142  042765  177400  002522                                                     MOV     (R2),TS4
6267  022150                                                                             BIC     #177400,
6268  022150  026527  003254  000001                IF PASCNT(R5) EQ #1 THEN      ;IF THIS IS PASS 1 THEN:
6269  022156  001014                                                                    CMP     PASCNT(R
6270  022160                                                                             BNE     50435$
6271  022160  016546  002522                          PRINTF #CODELM,DEVTBL(R5),TS4CL(R5) ;PRINT THE TS04 MICROCODE LEVEL.
6272  022164  016546  002532                                                           MOV     TS4CL(R5
6273  022170  012746  003772                                                           MOV     DEVTBL(R
6274  022174  012746  000003                                                           MOV     #CODELM,
6275  022200  010600                                                                   MOV     #3,-(SP)
6276  022202  104417                                                                   MOV     SP,R0
6277  022204  062706  000010                                                           TRAP    C$PNTF
6278  022210                                                                           ADD     #10,SP
6279  022210                                  ENDIF
6280  022210  004737  015520                                                50435$:
6281  022214                                  JSR   PC,NEXTU                    ;FIND NEXT UNIT.
6282  022214  000740                          ENDDO
6283  022216                                                                           BR      50433$
                                                                               50434$:
```

```
6284  022216                                    ENDSUB
6285  022216                              L10020:
6286  022216  104403                                                    TRAP    C$ESUB
6287
6288  022220                                    BGNSUB                ;SUBTEST 2   REWIND.
6289  022220                              T1.2:
6290  022220  104402                                                    TRAP    C$BSUB
6291
6292  022222                                    LET R2 := #BFSEQ1     ;ADR OF CMD SEQ.
6293  022222  012702  022774                                          MOV     #BFSEQ1,
6294  022226  004737  022676                    JSR     PC,BFSEQ      ;SET UP CMD SEQ.
6295  022232  004737  006510                    JSR     PC,EXALL      ;EXECUTE CMD SEQ ON ALL DEVICES.
6296  022236                                    LET STAFLG :B= #0     ;CLEAR START FLAG
6297  022236  105037  003452                                          CLRB    STAFLG
6298  022242                                    ENDSUB
6299  022242                              L10021:
6300  022242  104403                                                    TRAP    C$ESUB
6301
6302  022244                                    BGNSUB                ;SUBTEST 3   WRITE/VERIFY.
6303  022244                              T1.3:
6304  022244  104402                                                    TRAP    C$BSUB
6305
6306  022246                                    LET R2 := #BFSEQ2     ;ADR OF CMD SEQ.
6307  022246  012702  023006                                          MOV     #BFSEQ2,
6308  022252  004737  022676                    JSR     PC,BFSEQ      ;SET UP CMD SEQ.
6309  022256  004737  006510                    JSR     PC,EXALL      ;EXECUTE CMD SEQ ON ALL DEVICES.
6310  022262                                    ENDSUB
6311  022262                              L10022:
6312  022262  104403                                                    TRAP    C$ESUB
6313
6314  022264                                    BGNSUB                ;SUBTEST 4   WRITE TAPE MARK, ERASE.
6315  022264                              T1.4:
6316  022264  104402                                                    TRAP    C$BSUB
6317
6318  022266                                    LET R2 := #BFSEQ3     ;ADR OF CMD SEQ.
6319  022266  012702  023100                                          MOV     #BFSEQ3,
6320  022272  004737  022676                    JSR     PC,BFSEQ      ;SET UP CMD SEQ.
6321  022276  004737  006510                    JSR     PC,EXALL      ;EXECUTE CMD SEQ ON ALL DEVICES.
6322  022302                                    ENDSUB
6323  022302                              L10023:
6324  022302  104403                                                    TRAP    C$ESUB
6325
6326  022304                                    BGNSUB                ;SUBTEST 5 - SPACE FILES.
6327  022304                              T1.5:
6328  022304  104402                                                    TRAP    C$BSUB
6329
6330  022306                                    LET R2 := #BFSEQ4     ;ADR OF CMD SEQ.
6331  022306  012702  023152                                          MOV     #BFSEQ4,
6332  022312  004737  022676                    JSR     PC,BFSEQ      ;SET UP CMD SEQ.
6333  022316  004737  006510                    JSR     PC,EXALL      ;EXECUTE CMD SEQ ON ALL DEVICES.
6334  022322                                    ENDSUB
6335  022322                              L10024:
6336  022322  104403                                                    TRAP    C$ESUB
6337
6338  022324                                    BGNSUB                ;SUBTEST 6 - SPACE RECORDS.
6339  022324                              T1.6:
```

```
6340  022324  104402                                                    TRAP    C$BSUB
6341
6342  022326                                  LET R2 := #BFSEQ5          ;ADR OF CMD SEQ.
6343  022326  012702  023214                                            MOV     #BFSEQ5,
6344  022332  004737  022676                  JSR     PC,BFSEQ          ;SET UP CMD SEQ.
6345  022336  004737  006510                  JSR     PC,EXALL          ;EXECUTE CMD SEQ ON ALL DEVICES.
6346  022342                                  ENDSUB
6347  022342                          L10025:
6348  022342  104403                                                    TRAP    C$ESUB
6349
6350  022344                                  BGNSUB                    ;SUBTEST 7   WRITE RETRY.
6351  022344                          T1.7:
6352  022344  104402                                                    TRAP    C$BSUB
6353
6354  022346                                  LET R2 := #BFSEQ6          ;ADR OF CMD SEQ.
6355  022346  012702  023266                                            MOV     #BFSEQ6,
6356  022352  004737  022676                  JSR     PC,BFSEQ          ;SET UP CMD SEQ.
6357  022356  004737  006510                  JSR     PC,EXALL          ;EXECUTE CMD SEQ ON ALL DEVICES.
6358  022362                                  ENDSUB
6359  022362                          L10026:
6360  022362  104403                                                    TRAP    C$ESUB
6361
6362  022364                                  BGNSUB                    ;SUBTEST 8   READ REV RETRY.
6363  022364                          T1.8:
6364  022364  104402                                                    TRAP    C$BSUB
6365
6366  022366                                  LET R2 := #BFSEQ7          ;ADR OF CMD SEQ.
6367  022366  012702  023320                                            MOV     #BFSEQ7,
6368  022372  004737  022676                  JSR     PC,BFSEQ          ;SET UP CMD SEQ.
6369  022376  004737  006510                  JSR     PC,EXALL          ;EXECUTE CMD SEQ ON ALL DEVICES.
6370  022402                                  ENDSUB
6371  022402                          L10027:
6372  022402  104403                                                    TRAP    C$ESUB
6373
6374  022404                                  BGNSUB                    ;SUBTEST 9   READ FWD RETRY.
6375  022404                          T1.9:
6376  022404  104402                                                    TRAP    C$BSUB
6377
6378  022406                                  LET R2 := #BFSEQ8          ;ADR OF CMD SEQ.
6379  022406  012702  023352                                            MOV     #BFSEQ8,
6380  022412  004737  022676                  JSR     PC,BFSEQ          ;SET UP CMD SEQ.
6381  022416  004737  006510                  JSR     PC,EXALL          ;EXECUTE CMD SEQ ON ALL DEVICES.
6382  022422                                  ENDSUB
6383  022422                          L10030:
6384  022422  104403                                                    TRAP    C$ESUB
6385
6386  022424                                  BGNSUB                    ;SUBTEST 10  CLEAN.
6387  022424                          T1.10:
6388  022424  104402                                                    TRAP    C$BSUB
6389
6390  022426                                  LET R2 := #BFSEQ9          ;ADR OF CMD SEQ.
6391  022426  012702  023404                                            MOV     #BFSEQ9,
6392  022432  004737  022676                  JSR     PC,BFSEQ          ;SET UP CMD SEQ.
6393  022436  004737  006510                  JSR     PC,EXALL          ;EXECUTE CMD SEQ ON ALL DEVICES.
6394  022442                                  ENDSUB
6395  022442                          L10031:
```

```
6396  022442  104403                                                          TRAP    C$ESUB
6397
6398  022444                          BGNSUB                                ;SUBTEST 11 - WTV SWAPPED DATA BYTES.
6399  022444                  T1.11:
6400  022444  104402                                                          TRAP    C$BSUB
6401
6402  022446                          LET R2 := #BFSE10                     ;ADR OF CMD SEQ.
6403  022446  012702  023426                                                  MOV     #BFSE10,
6404  022452  004737  022676         JSR     PC,BFSEQ                      ;SET UP CMD SEQ.
6405  022456  004737  006510         JSR     PC,EXALL                      ;WRITE/VERIFY RECORDS 1 AND 2.
6406  022462                          LET SWBFLG :B  #1                     ;ENABLE BYTE SWAPPING.
6407  022462  112737  000001  003444                                        MOVB    #1,SWBFL
6408  022470  004737  006510         JSR     PC,EXALL                      ;WRITE/VERIFY RECORDS 3 AND 4.
6409  022474                          LET SWBFLG :B= #0                     ;DISABLE BYTE SWAPPING.
6410  022474  105037  003444                                                  CLRB    SWBFLG
6411  022500                          ENDSUB
6412  022500                  L10032:
6413  022500  104403                                                          TRAP    C$ESUB
6414
6415  022502                          LET R2 := DATAWT + #10.               ;INIT WRITE BUFFER POINTER.
6416  022502  013702  003334                                                  MOV     DATAWT,R
6417  022506  062702  000012                                                  ADD     #10.,R2
6418  022512                          WHILE R2 NE DATAWT DO   ;UNTIL 10 BYTES HAVE BEEN SWAPPED.
6419  022512                                                          50436$:
6420  022512  020237  003334                                                  CMP     R2,DATAW
6421  022516  001402                                                          BEQ     50437$
6422  022520  000342                    SWAB  (R2)                          ;SWAP DATA BYTES IN WRITE BUFFER.
6423  022522                          ENDDO
6424  022522  000773                                                          BR      50436$
6425  022524                                                          50437$:
6426  022524                          LET T1SWB :B= T1SWB + #1              ;SET T1 SWAP BYTES FLAG FOR "CKDATA" SUB
6427  022524  105237  003447                                                  INCB    T1SWB
6428
6429  022530                          BGNSUB                                ;SUBTEST 12 - READ SWAPPED DATA BYTES.
6430  022530                  T1.12:
6431  022530  104402                                                          TRAP    C$BSUB
6432
6433  022532                          LET CMDWRD := #RDR                    ;CMD IS READ REV.
6434  022532  012737  104401  003346                                        MOV     #RDR,CMD
6435  022540  004737  014504         JSR     PC,VFEXC                      ;VERIFY ODD LENGTH SWAP (RECORD 4).
6436  022544                          LET CMDPKT+CP.CNT :  #12              ;CHANGE BYTE COUNT TO 10.
6437  022544  012737  000012  002316                                        MOV     #12,CMDP
6438  022552  004737  014504         JSR     PC,VFEXC                      ;VERIFY EVEN LENGTH SWAP (RECORD 3).
6439  022556                          LET SWBFLG :B= #1                     ;ENABLE BYTE SWAPPING.
6440  022556  112737  000001  003444                                        MOVB    #1,SWBFL
6441  022564                          LET CMDPKT+CP.CNT := #11                ;CHANGE BYTE COUNT TO 9.
6442  022564  012737  000011  002316                                        MOV     #11,CMDP
6443  022572  004737  014504         JSR     PC,VFEXC                      ;VERIFY ODD LENGTH SWAP (RECORD 2).
6444  022576                          LET CMDPKT+CP.CNT := #12              ;CHANGE BYTE COUNT TO 10.
6445  022576  012737  000012  002316                                        MOV     #12,CMDP
6446  022604  004737  014504         JSR     PC,VFEXC                      ;VERIFY EVEN LENGTH SWAP (RECORD 1).
6447  022610                          LET CMDWRD := #RDF                    ;CMD IS READ FWD.
6448  022610  012737  104001  003346                                        MOV     #RDF,CMD
6449  022616  004737  014504         JSR     PC,VFEXC                      ;VERIFY EVEN LENGTH SWAP (RECORD 1).
6450  022622                          LET CMDPKT+CP.CNT := #11                ;CHANGE BYTE COUNT TO 9.
6451  022622  012737  000011  002316                                        MOV     #11,CMDP
```

```
  6452  022630  004737  014504                        JSR     PC,VFEXC              ;VERIFY ODD LENGTH SWAP (RECORD 2).
  6453  022634                                        LET SWBFLG :B= #0             ;DISABLE BYTE SWAPPING.
  6454  022634  105037  003444                                                      CLRB    SWBFLG
  6455  022640                                        LET CMDPKT+CP.CNT := #12      ;CHANGE BYTE COUNT TO 10.
  6456  022640  012737  000012  002316                                             MOV     #12,CMDP
  6457  022646  004737  014504                        JSR     PC,VFEXC              ;VERIFY EVEN LENGTH SWAP (RECORD 3).
  6458  022652                                        LET CMDPKT+CP.CNT := #11      ;CHANGE BYTE COUNT TO 9.
  6459  022652  012737  000011  002316                                             MOV     #11,CMDP
  6460  022660  004737  014504                        JSR     PC,VFEXC              ;VERIFY ODD LENGTH SWAP (RECORD 4).
  6461
  6462  022664                                        ENDSUB
  6463  022664                                L10033:
  6464  022664  104403                                                             TRAP    C$ESUB
  6465
  6466  022666                                        LET T1SWB :B= #0              ;CLEAR T1 SWAP BYTES FLAG
  6467  022666  105037  003447                                                     CLRB    T1SWB
  6468
  6469
  6470  022672                                        EXIT    TST
  6471  022672  104432                                                             TRAP    C$EXIT
  6472  022674  000554                                                             .WORD   L10017 .
```

```
6473                                        ;       SUBROUTINE TO MOVE A COMMAND SEQUENCE TO THE SEQUENCE TABLE.
6474                                        ;       INPUTS:         R2 = FWA OF COMMAND SEQUENCE.
6475                                        ;       OUTPUTS:
6476                                        ;       REGISTERS:
6477                                        ;       CALLS:
6478
6479  022676                                BFSEQ:  LET R1 := #CMDSEQ               ;INIT SEQ TABLE ADDRESS.
6480  022676  012701  003460                                                                  MOV     #CMDSEQ,
6481  022702                                        WHILE (R2) NE #END DO             ;WHILE THERE ARE MORE COMMANDS:
6482  022702                                                                                 50440$:
6483  022702  021227  177777                                                                 CMP     (R2),#EN
6484  022706  001402                                                                         BEQ     50441$
6485  022710                                          LET (R1)+ := (R2)+              ;MOVE COMMANDS TO SEQ TABLE.
6486  022710  012221                                                                         MOV     (R2)+,(R
6487  022712                                        ENDDO
6488  022712  000773                                                                         BR      50440$
6489  022714                                                                                 50441$:
6490  022714                                        LET (R1) := #END                  ;STORE END OF SEQUENCE CODE.
6491  022714  012711  177777                                                                 MOV     #END,(R1
6492  022720  000207                                RTS     PC                        ;RETURN.
6493
6494
6495
6496                                        ;       BASIC FUNCTION COMMAND SEQUENCE
6497
6498  022722  140004                        BFSEQ0: .WORD   SCH                       ;SET CHAR. 200.         (1)
6499  022724  000200                                        200
6500  022726  000001                                        1
6501  022730  000000                                        0
6502  022732  100013                                        DRI                       ;DRIVE INIT.            (2)
6503  022734  000001                                        1
6504  022736  000001                                        1
6505  022740  000000                                        0
6506  022742  140004                                        SCH                       ;SET CHAR. 20           (3)
6507  022744  000020                                        20
6508  022746  000001                                        1
6509  022750  000000                                        0
6510  022752  100017                                        GES                       ;GET STATUS.            (4)
6511  022754  000001                                        1
6512  022756  000001                                        1
6513  022760  000000                                        0
6514  022762  140004                                        SCH                       ;SET CHAR. 40.          (5)
6515  022764  000040                                        40
6516  022766  000001                                        1
6517  022770  000000                                        0
6518  022772  177777                              .WORD     END
6519
6520  022774  102010                        BFSEQ1:         RWD                       ;REWIND TWICE.          (6)
6521  022776  000001                                        1
6522  023000  000002                                        2
6523  023002  000000                                        0
6524  023004  177777                              .WORD     END
6525
6526  023006  104105                        BFSEQ2:         WTV                       ;WRITE/VERIFY PAT 1.    (7)
6527  023010  004000                                        DATCNT
6528  023012  000001                                        1
```

```
6529  023014  000001                                  1
6530  023016  104105                                  WTV             ;WTV PAT 2.             (8)
6531  023020  004000                                  DATCNT
6532  023022  000001                                  1
6533  023024  000002                                  2
6534  023026  104105                                  WTV             ;WTV PAT 3.             (9)
6535  023030  004000                                  DATCNT
6536  023032  000001                                  1
6537  023034  000003                                  3
6538  023036  104105                                  WTV             ;WTV PAT 4.             (10)
6539  023040  004000                                  DATCNT
6540  023042  000001                                  1
6541  023044  000004                                  4
6542  023046  104105                                  WTV             ;WTV PAT 5.             (11)
6543  023050  004000                                  DATCNT
6544  023052  000001                                  1
6545  023054  000005                                  5
6546  023056  104105                                  WTV             ;WTV PAT 6.             (12)
6547  023060  004000                                  DATCNT
6548  023062  000001                                  1
6549  023064  000006                                  6
6550  023066  104105                                  WTV             ;WTV PAT 0.             (13)
6551  023070  004000                                  DATCNT
6552  023072  000001                                  1
6553  023074  000000                                  0
6554  023076  177777                          .WORD   END
6555
6556  023100  100011                  BFSEQ3:         WTM             ;WRITE TAPE MARK.       (14)
6557  023102  000001                                  1
6558  023104  000001                                  1
6559  023106  000000                                  0
6560  023110  104005                                  WRT             ;WRITE 10 RECORDS.      (15)
6561  023112  004000                                  DATCNT
6562  023114  000010                                  10
6563  023116  000001                                  1
6564  023120  100411                                  ERS             ;ERASE 10 TIMES.        (16)
6565  023122  000001                                  1
6566  023124  000010                                  10
6567  023126  000000                                  0
6568  023130  100011                                  WTM             ;WRITE TAPE MARK.       (17)
6569  023132  000001                                  1
6570  023134  000001                                  1
6571  023136  000000                                  0
6572  023140  101011                                  WTR             ;WTM RETRY              (18)
6573  023142  000001                                  1
6574  023144  000001                                  1
6575  023146  000000                                  0
6576  023150  177777                          .WORD   END
6577
6578  023152  105410                  BFSEQ4:         SFR             ;SPACE 2 FILES REV.     (19)
6579  023154  000002                                  2
6580  023156  000001                                  1
6581  023160  000000                                  0
6582  023162  105010                                  SFF             ;SPACE 2 FILES FWD.     (20)
6583  023164  000002                                  2
6584  023166  000001                                  1
```

```
6585  023170  000000                                  0
6586  023172  105410                                  SFR             ;SPACE 2 FILES REV.     (21)
6587  023174  000001                                  1
6588  023176  000002                                  2
6589  023200  000000                                  0
6590  023202  105010                                  SFF             ;SPACE 2 FILES FWD.     (22)
6591  023204  000001                                  1
6592  023206  000002                                  2
6593  023210  000000                                  0
6594  023212  177777                          .WORD   END
6595
6596  023214  102010                  BFSEQ5:         RWD             ;REWIND.                (23)
6597  023216  000001                                  1
6598  023220  000001                                  1
6599  023222  000000                                  0
6600  023224  104010                                  SRF             ;SPACE 7 RECORDS FWD.   (24)
6601  023226  000007                                  7
6602  023230  000001                                  1
6603  023232  000000                                  0
6604  023234  104410                                  SRR             ;SPACE 7 RECORDS REV.   (25)
6605  023236  000007                                  7
6606  023240  000001                                  1
6607  023242  000000                                  0
6608  023244  104010                                  SRF             ;SPACE 7 RECORDS FWD.   (26)
6609  023246  000001                                  1
6610  023250  000007                                  7
6611  023252  000000                                  0
6612  023254  104410                                  SRR             ;SPACE 7 RECORDS REV.   (27)
6613  023256  000001                                  1
6614  023260  000007                                  7
6615  023262  000000                                  0
6616  023264  177777                          .WORD   END
6617
6618  023266  102010                  BFSEQ6:         RWD             ;REWIND.                (28)
6619  023270  000001                                  1
6620  023272  000001                                  1
6621  023274  000000                                  0
6622  023276  104005                                  WRT             ;WRITE.                 (29)
6623  023300  004000                                  DATCNT
6624  023302  000001                                  1
6625  023304  000001                                  1
6626  023306  105005                                  WRR             ;WRITE RETRY.           (30)
6627  023310  004000                                  DATCNT
6628  023312  000001                                  1
6629  023314  000001                                  1
6630  023316  177777                          .WORD   END
6631
6632  023320  104401                  BFSEQ7:         RDR             ;READ REV.              (31)
6633  023322  004000                                  DATCNT
6634  023324  000001                                  1
6635  023326  000001                                  1
6636  023330  105401                                  RNR             ;READ NEXT REV.         (32)
6637  023332  004000                                  DATCNT
6638  023334  000001                                  1
6639  023336  000001                                  1
6640  023340  125401                                  RNF             ;READ NEXT FWD.         (33)
```

```
6641  023342  004000                              DATCNT
6642  023344  000001                              1
6643  023346  000001                              1
6644  023350  177777                      .WORD   END
6645
6646  023352  104001              BFSEQ8:         RDF             ;READ FWD.              (34)
6647  023354  004000                              DATCNT
6648  023356  000001                              1
6649  023360  000001                              1
6650  023362  105001                              RPF             ;READ PREVIOUS FWD.     (35)
6651  023364  004000                              DATCNT
6652  023366  000001                              1
6653  023370  000001                              1
6654  023372  125001                              RPR             ;READ PREVIOUS REV.     (36)
6655  023374  004000                              DATCNT
6656  023376  000001                              1
6657  023400  000001                              1
6658  023402  177777                      .WORD   END
6659
6660  023404  101012              BFSEQ9: .WORD   CLN             ;CLEAN.                 (37)
6661  023406  000001                              1
6662  023410  000001                              1
6663  023412  000000                              0
6664  023414  102010                              RWD             ;REWIND                 (38)
6665  023416  000001                              1
6666  023420  000001                              1
6667  023422  000000                              0
6668  023424  177777                      .WORD   END             ;END OF SEQUENCE.
6669
6670  023426  104105              BFSE10:         WTV             ;WRITE/VERIFY EVEN LENGTH.      (39)
6671  023430  000012                              12
6672  023432  000001                              1
6673  023434  000000                              0
6674  023436  104105                              WTV             ;WRITE/VERIFY ODD LENGTH.       (40)
6675  023440  000011                              11
6676  023442  000001                              1
6677  023444  000000                              0
6678  023446  177777                      .WORD   END
6679                                      .EVEN
6680
6681  023450                              ENDTST
6682  023450                      L10017:
6683  023450  104401                                                      TRAP    CSETST
```

```
6684                                    .SBTTL  TEST 2: DATA RELIABILITY.
6685
6686                                    ;**
6687                                    ; TEST TO CHECK THE DATA RELIABILITY OF THE TS04.
6688                                    ;
6689  023452                                    BGNTST
6690  023452                            T2::
6691
6692  023452                                    LET RANDOM :B= #1                 ;SET THE RANDOM OPERATIONS FLAG.
6693  023452  112737  000001  003441                                                      MOVB    #1,RANDO
6694  023460                                    LET EXPBOT :B= #0                 ;CLEAR EXPECT BOT FLAG.
6695  023460  105037  003440                                                      CLRB    EXPBOT
6696  023464                                    LET R2 := #DATCNT   #1            ;SET UP THE RECORD LENGTH MASK.
6697  023464  012702  004000                                                      MOV     #DATCNT,
6698  023470  005302                                                             DEC     R2
6699  023472                                    LET LENMSK := COMP R2             ;ALLOW MAXIMUM BUFFER.
6700  023472  010237  003356                                                      MOV     R2,LENMS
6701  023476  005137  003356                                                      COM     LENMSK
6702  023502  004737  006444                  JSR PC,SETCH                      ;CMD 1 = SET CHARACTERISTIC.
6703  023506                                  IFB STAFLG NE #0 THEN             ;IF STARTING THEN:
6704  023506  105737  003452                                                      TSTB    STAFLG
6705  023512  001404                                                             BEQ     504423
6706  023514  004737  006470                    JSR   PC,SETRW                  ;CMD2=REWIND
6707  023520                                    LET STAFLG :B= #0                 ;CLR START FLAG.
6708  023520  105037  003452                                                      CLRB    STAFLG
6709  023524                                  ENDIF
6710  023524                                                                  504423:
6711  023524                                    LET (R1)+ := #WTV                 ;CMD3 = WRITE/ VERIFY.
6712  023524  012721  104105                                                      MOV     #WTV,(R1
6713  023530                                    LET (R1)+ := #DATCNT              ;SET BRF TO MAX FOR PATTERN GENERATION.
6714  023530  012721  004000                                                      MOV     #DATCNT,
6715  023534                                    LET R2 := COMP #RNOPSC
6716  023534  012702  177740                                                      MOV     #RNOPSC,
6717  023540  005102                                                             COM     R2
6718  023542                                  LET (R1)+ := R2                   ;31 OPERATIONS.
6719  023542  010221                                                             MOV     R2,(R1)+
6720  023544                                  LET (R1)+ := #RANP                ;RANDOM PATTERN.
6721  023544  012721  000007                                                      MOV     #RANP,(R
6722  023550                                  REPEAT                            ;REPEAT TO EOT:
6723  023550                                                                  50443$:
6724  023550                                    WHILE R1 LT #SEQEND DO          ;FILL SEQ TBL WITH RANDOM CMDS.
6725  023550                                                                  50444$:
6726  023550  020127  003550                                                      CMP     R1,#SEQE
6727  023554  002012                                                             BGE     50445$
6728  023556                                    LET RANS := RANS + RANB
6729  023556  063737  003360  003362                                              ADD     RANB,RAN
6730  023564                                    LET R2 := RANS CLR.BY #177741   ;R2 = RANDOM # (0   36).
6731  023564  013702  003362                                                      MOV     RANS,R2
6732  023570  042702  177741                                                      BIC     #1[illegible]41,
6733  023574  004772  023732                    JSR PC,@RANCMD(R2)              ;SET UP A RANDOM CMD + BRF.
6734  023600                                    ENDDO
6735  023600  000763                                                             BR      50444$
6736  023602                                                                  50445$:
6737  023602                                    LET (R1) := #END                ;STORE END OF SEQUENCE CODE IN TABLE.
6738  023602  012711  177777                                                      MOV     #END,,R1
6739  023606  004737  006510                    JSR   PC,EXALL                  ;GO EXECUTE ALL CMDS IN SEQUENCE TABLE.
```

```
023612                                  LET R1 := @CMDSEQ               ;INIT CMD SEQ TBL POINTER.
023612  012701  003460                                                      MOV     @CMDSEQ,
023616                                  UNTIL R2 NE @0                  ;REPEAT UNTIL EOT IS REACHED
023616  005702                                                            TST     R2
023620  001753                                                            BEQ     504435
023622                                  LET ALLEOT :B= ALLEOT + @1      ;FLAG ALL UNITS @ EOT
023622  105237  003450                                                    INCB    ALLEOT
023626  000240                          NOP
023630  000240                          NOP
023632  000240                          NOP
023634  004737  025156                  JSR     PC,T5WEOT               ;WRITE ONE RECORD BEYOND EOT ON ALL UNIT
                                                                        ;SO THAT SHORTER READ STOP DISTANCE
                                                                        ;SMALL POSITION HEAD IN CLEAN IRG GAP
                                                                        ;READ REV THAT EXTRA REC TO RE POSITION
023640  004737  023772                  JSR     PC,RANRD                ;SET UP READ REV/FWD CMDS.
023644                                  LET CMDSEQ+4 := COMP @RNOPSC    ;# OF RECORDS FOR READ REV.
023644  012737  177740  003464                                            MOV     @RNOPSC,
023652  005137  003464                                                    COM     CMDSEQ+4
023656                                  LET CMDSEQ+14 := CMDSEQ+4       ;# OF RECORDS FOR READ FORWARD.
023656  013737  003464  003474                                            MOV     CMDSEQ+4
023664                                  LET (R1) := @END                ;STORE END OF SEQUENSE CODE IN SEQ TABLE
023664  012711  177777                                                    MOV     @END,(R1
023670  004737  006510                  JSR     PC,EXALL                ;GO EXECUTE READ REV/FWD OF LAST N RECOR
023674                                  LET ALLEOT :B= @0               ;CLEAR ALL UNITS @ EOT FLAG
023674  105037  003450                                                    CLRB    ALLEOT
023700                                  LET RPTFLG :B= @1               ;REQUEST PERFORMANCE REPORT DURING REWIN
023700  112737  000001  003443                                            MOVB    @1,RPTFL
023706                                  LET R1 := @CMDSEQ               ;INIT SEQ TBL POINTER.
023706  012701  003460                                                    MOV     @CMDSEQ,
023712  004737  006470                  JSR PC,SETRW                    ;STORE REWIND IN SEQ TBL.
023716                                  LET (R1) := @END                ;STORE END IN SEQ TBL.
023716  012711  177777                                                    MOV     @END,(R1
023722  004737  006510                  JSR PC,EXALL                    ;EXECUTE REWIND CMD ON ALL UNITS
023726
023726  104432                          EXIT    TST
023730  000174                                                          TRAP    CSEXIT
                                                                        .WORD   L10034
```

```
                                        ;       ADDRESSES OF SUBROUTINES USED TO SET UP RANDOM OPERATIONS IN
                                        ;       THE DATA RELIABILITY TEST.
                                        ;
023732  024060                          RANCMD: RANWV                   ;WRITE/VERIFY.
023734  024046                                  RANWR                   ;WRITE.
023736  024046                                  RANWR                   ;WRITE.
023740  024046                                  RANWR                   ;WRITE.
023742  024046                                  RANWR                   ;WRITE.
023744  024046                                  RANWR                   ;WRITE.
023746  024046                                  RANWR                   ;WRITE.
023750  024046                                  RANWR                   ;WRITE.
023752  023772                                  RANRD                   ;READ.
023754  023772                                  RANRD                   ;READ.
023756  023772                                  RANRD                   ;READ.
023760  023772                                  RANRD                   ;READ.
023762  023772                                  RANRD                   ;READ.
023764  023772                                  RANRD                   ;READ.
023766  023772                                  RANRD                   ;READ.
023770  023772                                  RANRD                   ;READ.
                                        ;       SUBROUTINE TO SET UP READ COMMANDS IN SEQUENCE TABLE.
                                        ;       INPUTS:
                                        ;       OUTPUTS:
                                        ;       REGISTERS:      R2
                                        ;       CALLS:
023772                                  RANRD:  LET (R1)+ := @RDR               ;STORE READ REV CMD.
023772  012721  104401                                                                  MOV     @RDR,(R1
023776                                  LET (R1)+ := @DATCNT            ;SET BRF TO MAX FOR READ RANDOM LENGTHS.
023776  012721  004000                                                                  MOV     @DATCNT,
024002                                  LET RANB := RANB + RAN,
024002  063737  003362  003360                                                         ADD     RANS,RAN
024010                                  LET R2 := RANB CLR.B+ @RI.OPSC
024010  013702  003360                                                                 MOV     RANB,R2
024014  042702  177740                                                                 BIC     @RNOPSC,
024020                                  LET (R1)+ := R2                 ;SET RANDOM # OF OPERATIONS.
024020  010221                                                                         MOV     R2,(R1)+
024022                                  LET (R1)+ := @RANP              ;RANDOM PATTERN.
024022  012721  000007                                                                 MOV     @RANP,(R
024026                                  LET (R1)+ := @RDF               ;STORE READ FWD CMD.
024026  012721  104001                                                                 MOV     @RDF,(R1
024032                                  LET (R1)+ := @DATCNT            ;SET BRF TO MAX TO READ RANDOM LENGTHS.
024032  012721  004000                                                                 MOV     @DATCNT,
024036                                  LET (R1)+ := R2                 ;SET RANDOM # OF OPERATIONS.
024036  010221                                                                         MOV     R2,(R1)+
024040                                  LET (R1)+ := @RANP              ;RANDOM PATTERN.
024040  012721  000007                                                                 MOV     @RANP,(R
024044  000207                          RTS PC
```

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

024046                                  RANWR:  LET (R1)+ := #WRT               ;STORE WRITE CMD.
024046  012721  104005                                                                  MOV     #WRT,(R1
024052  004737  024072                          JSR PC,RANW                       ;STORE BRF, # OF OPERATIONS, PATTERN.
024056  000207                                  RTS PC


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

024060                                  RANWV:  LET (R1)+ := #WTV               ;STORE WRITE/VERIFY CMD.
024060  012721  104105                                                                  MOV     #WTV,(R1
024064  004737  024072                          JSR PC,RANW                       ;STORE BRF, # OF OPERATIONS, PATTERN.
024070  000207                                  RTS     PC
```

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

024072                  RANW:   LET (R1)+ := #DATCNT            ;SET BRF TO MAX FOR PATTERN GENERATION.
024072  012721  004000                                                  MOV     #DATCNT,
                                                                ;RANDOM BRF WILL BE GENERATED FOR EACH R
024076                          LET RANB := RANB + RANS
024076  063737  003362  003360                                          ADD     RANS,RAN
024104                          LET R2 := RANB CLR.BY #RNOPSC
024104  013702  003360                                                  MOV     RANB,R2
024110  042702  177740                                                  BIC     #RNOPSC,
024114                          LET (R1)+ :  R2                 ;SET RANDOM # OF OPERATIONS.
024114  010221                                                          MOV     R2,(R1)+
024116                          LET (R1)+ := #RANP              ;RANDOM PATTERN.
024116  012721  000007                                                  MOV     #RANP,(R
024122  000207                  RTS PC                          ;RETURN.
                                .EVEN
                                ENDTST
024124                  L10034:
024124  104401                                                          TRAP    C$ETST
```

```
                                        .SBTTL  TEST 3: WRITE COMPATABILITY/WRITE UTILITY.

                                        ;**
                                        ; TEST TO WRITE RECORDS FROM BOT TO EOT.
                                        ;

                                                BGNTST
                                        T3::

                                                LET RANDOM :B= #1               ;SET THE RANDOM OPERATIONS FLAG.
024126                                                                                  MOVB    #1,RANDO
024126  112737  000001  003441
                                                LET EXPBOT :B= #0               ;CLEAR EXPECT BOT FLAG.
024134                                                                                  CLRB    EXPBOT
024134  105037  003440
                                                LET R2 := #DATCNT - #1          ;SET UP THE RECORD LENGTH MASK.
024140                                                                                  MOV     #DATCNT,
024140  012702  004000                                                                  DEC     R2
024144  005302
                                                LET LENMSK := COMP R2           ;ALLOW MAXIMUM BUFFER.
024146                                                                                  MOV     R2,LENMS
024146  010237  003356                                                                  COM     LENMSK
024152  005137  003356
                                                JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
024156  004737  006444                          JSR PC,SETRW                    ;CMD2=REWIND
024162  004737  006470                          LET STAFLG :B= #0               ;CLEAR START FLAG
024166                                                                                  CLRB    STAFLG
024166  105037  003452
                                                REPEAT                          ;REPEAT TO EOT.
024172                                                                                  50446$:
024172                                            WHILE R1 LT #SEQEND DO        ;WHILE THERE IS MORE ROOM IN SEQ TABLE:
024172                                                                                  50447$:
024172                                                                                  CMP     R1,#SEQE
024176  020127  003550                                                                  BGE     50450$
024200  002003
024204  004737  024046                              JSR PC,RANWR                    ;STORE A WRITE CMD IN SEQUENCE TABLE.
                                                ENDDO
024204
024204  000772                                                                        BR      50447$
024206                                                                                  50450$:
024206                                          LET (R1) := #END                ;STORE END OF SEQUENCE CODE IN TABLE.
024206  012711  177777                                                                  MOV     #END,(R1
024212  004737  006510                          JSR   PC,EXALL                  ;EXECUTE ALL CMDS IN SEQ TBL ON UNITS.
024216                                          LET R1 := #CMDSEQ               ;INIT SEQ TBL POINTER.
024216  012701  003460                                                                  MOV     #CMDSEQ,
024222                                        UNTIL R2 NE #0                    ;REPEAT UNTIL EOT IS REACHED
024222  005702                                                                          TST     R2
024224  001762                                                                        BEQ     50446$
024226
                                        LET ALLEOT :B= ALLEOT + #1      ;SET ALL UNITS @ EOT FLAG
024226  105237  003450                                                                  INCB    ALLEOT
024232  000240                        NOP
024234  000240                        NOP
024236  000240                        NOP
024240  004737  025156                JSR     PC,T5WEOT               ;WRITE ONE RECORD BEYOND EOT ON ALL UNIT
                                                                        ;SO THAT SHORTER READ STOP DISTANCE
                                                                        ;SHALL POSITION HEAD IN CLEAN IRG GAP
                                                                        ;READ REV THAT EXTRA REC TO RE POSITION
024244                          LET ALLEOT :B= #0               ;CLEAR ALL UNITS @ EOT FLAG
024244  105037  003450                                                                  CLRB    ALLEOT
024250  004737  006470          JSR PC,SETRW                    ;STORE REWIND IN SEQ TBL.
024254                          LET (R1) := #END                ;STORE END IN SEQ TBL.
024254  012711  177777                                                                  MOV     #END,(R1
024260  004737  006510          JSR PC,EXALL                    ;EXECUTE REWIND CMD ON ALL UNITS
```

```
6942
6943  024264                                  EXIT    TST
6944  024264  104432                                                  TRAP    C$EXIT
6945  024266  000002                                                  .WORD   L10035-.
6946
6947                                          .EVEN
6948
6949  024270                                  ENDTST
6950  024270                            L10035:
6951  024270  104401                                                  TRAP    C$ETST
6952
```

```
                                        .SBTTL  TEST 4:  READ COMPATABILITY/READ UTILITY.

                                        ;++
                                        ; TEST TO READ ENTIRE TAPE FORWARD AND REVERSE.
                                        ;

                                                BGNTST
                                        T4::

                                                LET RANDOM :B- #1               ;SET THE RANDOM OPERATIONS FLAG.
024272  112737  000001  003441                                                          MOVB    #1,RANDO
                                                LET EXPBOT :B= #1               ;SET EXPECT BOT FLAG.
024300  112737  000001  003440                                                          MOVB    #1,EXPBO
                                                JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
                                                JSR PC,SETRW                    ;CMD2=REWIND.
                                                LET STAFLG :B= #0               ;CLEAR START FLAG
024316  105037  003452                                                          CLRB    STAFLG
                                                LET (R1)+ := #RDF               ;CMD3 = READ FORWARD.
024322  012721  104001                                                          MOV     #RDF,(R1
                                                LET (R1)+ := #DATCNT            ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
024326  012721  004000                                                          MOV     #DATCNT,
                                                LET (R1)+ := #77777             ;SET RECORD COUNT TO MAX FOR WHOLE TAPE.
024332  012721  077777                                                          MOV     #77777,(
                                                LET (R1)+ := #RANP              ;PATTERN = RANDOM.
024336  012721  000007                                                          MOV     #RANP,(R
                                                LET (R1) := #END                ;STORE END OF SEQUENCE CODE IN TABLE.
024342  012711  177777                                                          MOV     #END,(R1
                                                JSR     PC,EXALL                ;EXECUTE ALL CMDS IN SEQ TBL ON ALL UNIT
                                                LET ALLEOT :B= ALLEOT + #1      ;FLAG TO ALLOW ALL UNITS AT EOT TO READ
024352  105237  003450                                                          INCB    ALLEOT
                                                LET R1 := #CMDSEQ               ;INIT CMD SEQ TBL POINTER.
024356  012701  003460                                                          MOV     #CMDSEQ,
                                                LET (R1)+ := #RDR               ;CMD1 = READ REVERSE.
024362  012721  104401                                                          MOV     #RDR,(R1
                                                LET (R1)+ := #DATCNT            ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
024366  012721  004000                                                          MOV     #DATCNT,
                                                LET (R1)+ := #77777             ;RECORD COUNT = MAX FOR WHOLE TAPE.
024372  012721  077777                                                          MOV     #77777,(
                                                LET (R1)+ := #RANP              ;PATTERN = RANDOM.
024376  012721  000007                                                          MOV     #RANP,(R
                                                LET (R1) := #END                ;STORE END OF SEQUENCE CODE IN TABLE.
024402  012711  177777                                                          MOV     #END,(R1
                                                JSR     PC,EXALL                ;GO EXECUTE READ REV. OF ENTIRE TAPE.
                                                LET ALLEOT :B= #0               ;CLEAR ALL UNITS @ EOT FLAG
024412  105037  003450                                                          CLRB    ALLEOT

                                                EXIT    TST
024416  104432                                                                  TRAP    C$EXIT
024420  000002                                                                  .WORD   L10036-.

                                                .EVEN

                                                ENDTST
024422                                  L10036:
024422  104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL  TEST 5: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.

                                ;**
                                ; TEST TO EXECUTE  OPERATOR SELECTED COMMAND SEQUENCE.
                                ;

                                        BGNTST
                                T5::

                                        LET RANDOM :B  #0               ;CLEAR RAMDOM MODE FLAG.
024424  105037  003441                                                          CLRB    RANDOM
                                        LET EXPBOT :B= #1               ;SET EXPECT BOT FLAG.
024430  112737  000001  003440                                                 MOVB    #1,EXPBO
                                        LET IRE :B= PIRE                ;MOVE INHIBIT RFC ERROR REPORT FLAG.
024436  113737  002214  003445                                                 MOVB    PIRE,IRE
                                        JSR PC,SETCH                    ;CMD 1 = SET CHARACTERISTIC.
024444  004737  006444
                                        LET CMDSEQ+2 := CHAR            ;MOVE CHAR CODE FROM P TBL TO SEQ TBL.
024450  013737  002216  003462                                                 MOV     CHAR,CMD
                                        LET R2 := #CMDD                 ;R2 POINTS TO CMD2 IN SOFT P TABLE.
024456  012702  002220                                                          MOV     #CMDD,R2
                                        JSR     PC,PTCMDS               ;MOVE CMD 2 FROM P TBL TO SEQ TBL.
024462  004737  025134                  JSR     PC,PTCMDS               ;MOVE CMD 3 FROM P TBL TO SEQ TBL.
024466  004737  025134                  JSR     PC,PTCMDS               ;MOVE CMD 4 FROM P TBL TO SEQ TBL.
024472  004737  025134                  JSR     PC,PTCMDS               ;MOVE CMD 5 FROM P TBL TO SEQ TBL.
024476  004737  025134                  JSR     PC,PTCMDS               ;MOVE CMD 6 FROM P TBL TO SEQ TBL.
024502  004737  025134                  JSR     PC,PTCMDS               ;MOVE CMD 7 FROM P TBL TO SEQ TBL.
024506  004737  025134                  JSR     PC,PTCMDS               ;MOVE END CMD FROM P TBL TO SEQ TBL.
024512  004737  025134                  LET JLOOP := #0                 ;CLEAR JMP CMD LOOP COUNT.
024516  005037  003370                                                          CLR     JLOOP
                                        LET STAFLG :B= #0               ;CLEAR START FLAG
024522  105037  003452                                                          CLRB    STAFLG
                                        LET R1 := #CMDSEQ               ;INIT SEQUENCE TABLE POINTER.
024526  012701  003460                                                          MOV     #CMDSEQ,
                                3$:     WHILE (R1) NE #END DO           ;WHILE THERE ARE CMDS LEFT IN SEQUENCE *
024532                                                                                  50451$:
024532  021127  177777                                                          CMP     (R1),#EN
024536  001574                                                                  BEQ     50452$
024540  022711  000040                    CMP   #JMP.C,(R1)               ;IS THIS A JUMP CMD?
024544  001024                            BNE   6$                        ;BR IF NOT.
                                          LET R1 := R1 + #2               ;POINT TO BRF.
024546  062701  000002                                                          ADD     #2,R1
024552  012137  0033'2                   MOV   (R1)+,JLOC                ;SAVE BRF (LOCATION).
024556  022137  003370                    CMP (R1)+,JLOOP                 ;HAS LOOP COUNT BE SATISFIED?
024562  001003                            BNE   1$                        ;IF NOT, JMP AGAIN.
                                          LET R1 := R1 + #2               ;IF SO, ADJUST SEQ POUNTER
024564  062701  000002                                                          ADD     #2,R1
024570  000760                            BR    3$                        ;AND GO TO NEXT COMMAND.
                                1$:       LET JLOOP := JLOOP + #1         ;UPDATE THE LOOP COUNT.
024572  005237  003370                                                          INC     JLOOP
                                          LET R1 := #CMDSEQ               ;INIT CMD SEQ TABLE POINTER.
024576  012701  003460                                                          MOV     #CMDSEQ,
                                2$:       DEC   JLOC                      ;DECR LOCATION COUNTER.
024602  005337  003372                    BEQ   3$                        ;IF THIS IS THE RIGHT LOCATION TO JMP TO
024606  001751                            LET R1 := R1 + #10              ;IF NOT, UPDATE SEQ POINTER TO NEXT CMD.
024610                                                                          ADD     #10,R1
024610  062701  000010                    BR    2$                        ;DO IT AGAIN.
024614  000772
```

```
7065  024616  022711  000020            6$:     CMP   #DLY.C,(R1)             ;DELAY?
7066  024622  001026                            BNE   4$                       ;BR IF NOT.
7067  024624                                    LET R1 := R1 + #4               ;R1 = LOCATION OF N COUNT.
7068  024624  062701  000004                                                            ADD     #4,R1
7069  024630                                    LET TIME2 := (R1)               ;SAVE N COUNT.
7070  024630  011137  003366                                                            MOV     (R1),TIM
7071  024634                                    7$:     DELAY 1                 ;GO TO SUPER WAIT 1 MSEC.
7072  024634  012727  000001                                                            MOV     #1,(PC)+
7073  024640  000000                                                                    .WORD   0
7074  024642  013727  002116                                                            MOV     L$DLY,(P
7075  024646  000000                                                                    .WORD   0
7076  024650  005367  177772                                                            DEC      6(PC)
7077  024654  001375                                                                    BNE     .-4
7078  024656  005367  177756                                                            DEC     -22(PC)
7079  024662  001367                                                                    BNE     .-20
7080  024664  005337  003366                    DEC   TIME2
7081  024670  001361                            BNE   7$
7082  024672                                    LET R1 := R1 + #4               ;POINT TO NEXT CMD.
7083  024672  062701  000004                                                            ADD     #4,R1
7084  024676  000715                           BR    3$                        ;GO CHECK NEXT CMD.
7085  024700  004737  007452            4$:     JSR   PC,SETUP                  ;GO SETUP THE COMMAND BLOCK.
7086  024704                                    WHILE NCNT LT NCNT1 DO          ;WHILE THERE ARE RECORDS REMAINING:
7087  024704                                                                                50453$:
7088  024704  023737  003340  003342                                                            CMP     NCNT,NCN
7089  024712  002103                                                                          BGE     50454$
7090  024714  004737  007344                      JSR PC,CMDAC                  ;STORE CMD ASCII IN ERROR MSG.
7091  024720  004737  007004                      JSR PC,EXSUB                  ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATU
7092  024724                                      IF CMDWRD EQ #GES THEN        ;IF CMD IS GET STATUS THEN:
7093  024724  023727  003346  100017                                                          CMP     CMDWRD,#
7094  024732  001002                                                                        BNE     50455$
7095  024734  004737  015776                        JSR PC,PRXST                ;PRINT EXTENDED STATUS REGISTERS.
7096  024740                                      ENDIF
7097  024740                                                                                50455$:
7098  024740  004737  016060                    JSR PC,CKHAE                  ;CHECK HALT AFTER EACH CMD FLAG.
7099  024744                                    LET R2 := #1                  ;SET ALL UNITS AT BOT/EOT.
7100  024744  012702  000001                                                            MOV     #1,R2
7101  024750  004737  015452                    JSR PC,FIRSTU                 ;FIND FIRST UNIT.
7102  024754                                    WHILE DEVTBL(R5) NE #END DO   ;WHILE THERE ARE MORE UNITS:
7103  024754                                                                                50456$:
7104  024754  026527  002532  177777                                                            CMP     DEVTBL(R
7105  024762  001426                                                                          BEQ     50457$
7106  024764                                      IF #MOD.CO SETIN CMDWRD THEN  ;IF CMD IS REVERSE THEN:
7107  024764  032737  000400  003346                                                            BIT     #MOD.CO.
7108  024772  001406                                                                          BEQ     50460$
7109  024774                                        IF #XO.BOT NOTSETIN EOTFLG(R5) THEN  ;IF NOT AT BOT THEN:
7110  024774  032765  000002  003426                                                            BIT     #XO.BOT.
7111  025002  001001                                                                          BNE     50461$
7112  025004                                          LET R2 := #0                ;CLEAR EOT/BOT FLAG.
7113  025004  005002                                                                          CLR     R2
7114  025006                                        ENDIF
7115  025006                                                                                50461$:
7116  025006                                      ELSE                          ;ELSE IF CMD IS NOT REVERSE:
7117  025006  000411                                                                          BR      50462$
7118  025010                                                                                50460$:
7119  025010                                        IF #XO.EOT NOTSETIN EOTFLG(R5) OR #CMD.CO NOTSETIN CMDWRD THEN
7120  025010  032765  000001  003426                                                            BIT     #XO.EOT.
```

```
7121  025016  001404                                                                    BEQ     50463$
7122  025020  032737  000001  003346                                                    BIT     #CMD.CO,
7123  025026  001001                                                                    BNE     50464$
7124  025030                                                                  50463$:
7125                                                                  ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
7126  025030                                  LET R2 := #0              ;CLEAR EOT/BOT FLAG.
7127  025030  005002                                                                   CLR     R2
7128  025032                                  ENDIF
7129  025032                                                                  50464$:
7130  025032                                ENDIF
7131  025032                                                                  50462$:
7132  025032  004737  015520                JSR PC,NEXTU              ;FIND NEXT UNIT
7133  025036                              ENDDO                       ;
7134  025036  000746                                                                   BR      50456$
7135  025040                                                                  50457$:
7136  025040                              IF R2 EQ #1 THEN            ;IF ALL UNIT ARE AT EOT/BOT THEN:
7137  025040  020227  000001                                                          CMP     R2,#1
7138  025044  001016                                                                  BNE     50465$
7139  025046                                LET NCNT1 := NCNT + #1    ;FORCE TERMINATION OF COMMAND.
7140  025046  013737  003340  003342                                                  MOV     NCNT,NCN
7141  025054  005237  003342                                                         INC     NCNT1
7142  025060                                LET ALLEOT :B= ALLEOT + #1 ;FLAG ALL UNITS AT EOT/BOT TO ALLOW VER
7143  025060  105237  003450                                                         INCB    ALLEOT
7144  025064                                IF CMDLG EQ #2 THEN       ;WHEN WRITING IS CURRENT COMMAND
7145  025064  023727  003354  000002                                                  CMP     CMDLG,#2
7146  025072  001002                                                                  BNE     50466$
7147  025074  004737  025156                  JSR PC,T5WEOT           ;GO WRITE/READ REV ONE RECORD BEYOND EOT
7148  025100                                ENDIF
7149  025100                                                                  50466$:
7150  025100                              ELSE
7151  025100  000402                                                                   BR      50467$
7152  025102                                                                  50465$:
7153  025102                                  LET ALLEOT :B- #0 ;WHEN NOT ALL @EOT, CLEAR FLAG
7154  025102  105037  003450                                                         CLRB    ALLEOT
7155  025106                                ENDIF
7156  025106                                                                  50467$:
7157  025106                                LET NCNT :- NCNT + #1     ;UPDATE RECORD COUNT.
7158  025106  005237  003340                                                         INC     NCNT
7159  025112                                LET PCMDWD :  CMDWRD      ;SAVE PREVIOUS COMMAND WORD.
7160  025112  013737  003346  003352                                                  MOV     CMDWRD,P
7161  025120                              ENDDO
7162  025120  000671                                                                   BR      50453$
7163  025122                                                                  50454$:
7164  025122  004737  014402                JSR   PC,VFYDAT           ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
7165                                                                  ;VERIFY THE LAST N RECORDS OF DATA.
7166  025126                              ENDDO
7167  025126  000601                                                                   BR      50451$
7168  025130                                                                  50452$:
7169
7170  025130                              EXIT    TST
7171  025130  104432                                                                  TRAP    C$EXIT
7172  025132  000130                                                                  .WORD   L1003-.
7173
7174
7175
7176
```

```
7177
7178
7179                                    ;       SUBROUTINE TO MOVE A COMMAND FROM THE SOFTWARE P TABLE TO
7180                                    ;       THE COMMAND SEQUENCE TABLE.
7181                                    ;       INPUTS:         R2 = POINTER TO SOFT 'P' TABLE
7182                                    ;       OUTPUTS:
7183                                    ;       REGISTERS:      R3.
7184                                    ;       CALLS:
7185
7186  025134                            PTCMDS: LET R3 := (R2)+   #1 SHIFT +1   ;R3 = COMMAND TABLE INDEX.
7187  025134  012203                                                                    MOV     (R2)+,R3
7188  025136  005303                                                                    DEC     R3
7189  025140  006303                                                                    ASL     R3
7190  025142                                    LET (R1)+ :- CMDTBL(R3)         ;MOVE COMMAND WORD.
7191  025142  016321  003562                                                           MOV     CMDTBL(R
7192  025146                                    LET (R1)+ := (R2)+              ;MOVE # OF BYTES.
7193  025146  012221                                                                   MOV     (R2)+,(R
7194  025150                                    LET (R1)+ := (R2)+              ;MOVE # OF OPERATIONS.
7195  025150  012221                                                                   MOV     (R2)+,(R
7196  025152                                    LET (R1)+ := (R2)+              ;MOVE PATTERN CODE.
7197  025152  012221                                                                   MOV     (R2)+,(R
7198  025154  000207                           RTS PC
```

```
7199
7200                                    ;       SUBROUTINE TO WRITE THEN READ REVERSE ONE RECORD BEYOND EOT
7201                                    ;       INPUTS:
7202                                    ;       OUTPUTS:
7203                                    ;       REGISTERS:
7204                                    ;       CALLS:          CMDAC,EXSUB,CKHAE
7205
7206  025156  000240                    TSWEOT: NOP
7207  025160  000240                            NOP
7208  025162  004737  007004                    JSR PC,EXSUB            ;WRITE ONE RECORD BEYOND EOT
7209  025166  004737  016060                    JSR PC,CKHAE            ;SO THAT READ SHORTER STOP DISTANCE
7210                                                                    ;SHALL POSITION HEAD IN CLEAN IRG GAP
7211  025172                                    LET PCMDWD := CMDWRD    ;REPOSITION TAPE
7212  025172  013737  003346  003352                                                            MOV     CMDWRD,P
7213  025200                                    LET CMDWRD := #RDR      ;BEFORE EXTRA RECORD
7214  025200  012737  104401  003346                                                            MOV     #RDR,CMD
7215  025206                                    LET CMDLG := #4         ;BY READING REVERSE
7216  025206  012737  000004  003354                                                            MOV     #4,CMDLG
7217  025214                                    LET CMDPKT := CMDWRD CLR.BY #BRF.C
7218  025214  013737  003346  002310                                                            MOV     CMDWRD,C
7219  025222  042737  004000  002310                                                            BIC     #BRF.C,C
7220  025230                                    LET CMDSAV := CMDPKT    ;THAT RECORD TO ALLOW
7221  025230  013737  002310  003350                                                            MOV     CMDPKT,C
7222  025236                                    LET CMDPKT+CP.ADL :- DATARD     ;NEXT COMMAND IN THE
7223  025236  013737  003336  002312                                                            MOV     DATARD,C
7224  025244  004737  007344                    JSR PC,CMDAC            ;TABLE TO BE EXECUTED
7225  025250  004737  007004                    JSR PC,EXSUB
7226  025254  004737  016060                    JSR PC,CKHAE
7227  025260  000207                            RTS PC
7228
7229
7230                                            .EVEN
7231
7232  025262                                    ENDTST
7233  025262                                L10037:
7234  025262  104401                                                           TRAP    CSETST
7235
7236  025264                                    ENDMOD
```

```
                                .TITLE PARAMETER CODING

                                .SBTTL  HARDWARE PARAMETER CODING SECTION

025264                                  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.
                                ;

025264                          BGNHRD
025264  000024                                                    .WORD   L10040-L$HARD
025266                          L$HARD::

025266                          GPRMA   TS4ADR,0,0,160000,177564,YES
025266  000031                                                    .WORD   T$CODE
025270  025312                                                    .WORD   TS4ADR
025272  160002                                                    .WORD   T$LOLIM
025274  177564                                                    .WORD   T$HILIM
025276                          GPRMD   TS4VCT,2,0,777,60,776,YES
025276  001032                                                    .WORD   T$CODE
025300  025327                                                    .WORD   TS4VCT
025302  000777                                                    .WORD   777
025304  000060                                                    .WORD   T$LOLIM
025306  000776                                                    .WORD   T$HILIM

025310                          EXIT HRD
025310  013004                                                    .WORD   T$CODE

                                        .NLIST  BEX
025312  051524  051123  040440  TS4ADR: .ASCIZ  /TSSR ADDRESS/
025327  000126  041505  047524  TS4VCT: .ASCIZ  /VECTOR/
                                        .LIST   BEX
                                        .EVEN

025336                          ENDHRD
                                                                    .EVEN
025336                          L10040:
```

```
                                        .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.
                                        ;

025536                                          BGNSFT
025536  000501                                                        .WORD L10041 L$S
025540                                  L$SOFT::

025540                                          GPRML   CLRM,0,1,YES
025540  000130                                                        .WORD   T$CODE
025542  026102                                                        .WORD   CLRM
025544  000001                                                        .WORD   1
025546                                          GPRML   RRVM,0,400,YES
025546  000130                                                        .WORD   T$CODE
025550  026121                                                        .WORD   RRVM
025552  000400                                                        .WORD   400
025554                                          GPRML   HAEM,2,1,YES
025554  001130                                                        .WORD   T$CODE
025556  026150                                                        .WORD   HAEM
025560  000001                                                        .WORD   1
025562                                          GPRML   RCVERM,2,400,YES
025562  001130                                                        .WORD   T$CODE
025564  026174                                                        .WORD   RCVERM
025566  000400                                                        .WORD   400
025570                                          GPRML   IRECM,4,1,YES
025570  002130                                                        .WORD   T$CODE
025572  026225                                                        .WORD   IRECM
025574  000001                                                        .WORD   1
025576                                          XFERT   NEXTSP
025576  004024                                                        .WORD   T$CODE
025600                                          GPRML   BADTM,4,400,YES
025600  002130                                                        .WORD   T$CODE
025602  026246                                                        .WORD   BADTM
025604  000400                                                        .WORD   400
025606                                  NEXTSP: GPRML   DINTM,6,1,YES
025606  003130                                                        .WORD   T$CODE
025610  026276                                                        .WORD   DINTM
025612  000001                                                        .WORD   1
025614                                          GPRML   IREM,6,400,YES
025614  003130                                                        .WORD   T$CODE
025616  026321                                                        .WORD   IREM
025620  000400                                                        .WORD   400
025622                                          GPRML   CHGM,10,1,YES
025622  004130                                                        .WORD   T$CODE
025624  026352                                                        .WORD   CHGM
025626  000001                                                        .WORD   1
025630                                          XFERF   ENDSP1
025630  127044                                                        .WORD   T$CODE
```

```
7333  025432                    GPRMD   CHARM,14,0,377,0,777,YES
7334  025432  006032                                    .WORD   T$CODE
7335  025434  026376                                    .WORD   CHARM
7336  025436  000377                                    .WORD   377
7337  025440  000000                                    .WORD   T$LOLIM
7338  025442  000777                                    .WORD   T$HILIM
7339  025444                    GPRMD   CMD2M,16,D,37,1,33,YES
7340  025444  007052                                    .WORD   T$CODE
7341  025446  026423                                    .WORD   CMD2M
7342  025450  000037                                    .WORD   37
7343  025452  000001                                    .WORD   T$LOLIM
7344  025454  000033                                    .WORD   T$HILIM
7345  025456                    GPRMD   BPCRM,20,D, 1,1,DATCNT,YES
7346  025456  010052                                    .WORD   T$CODE
7347  025460  026431                                    .WORD   BPCRM
7348  025462  177777                                    .WORD   -1
7349  025464  000001                                    .WORD   T$LOLIM
7350  025466  004000                                    .WORD   T$HILIM
7351  025470                    GPRMD   NUMBM,22,D, 1,1,77777,YES
7352  025470  011052                                    .WORD   T$CODE
7353  025472  026443                                    .WORD   NUMBM
7354  025474  177777                                    .WORD    1
7355  025476  000001                                    .WORD   T$LOLIM
7356  025500  077777                                    .WORD   T$HILIM
7357  025502                    GPRMD   PATTM,24,D,17,0,10,YES
7358  025502  012052                                    .WORD   T$CODE
7359  025504  026463                                    .WORD   PATTM
7360  025506  000017                                    .WORD   17
7361  025510  000000                                    .WORD   T$LOLIM
7362  025512  000010                                    .WORD   T$HILIM
7363  025514                    GPRMD   CMD3M,26,D,37,1,33,YES
7364  025514  013052                                    .WORD   T$CODE
7365  025516  026476                                    .WORD   CMD3M
7366  025520  000037                                    .WORD   37
7367  025522  000001                                    .WORD   T$LOLIM
7368  025524  000033                                    .WORD   T$HILIM
7369  025526                    GPRMD   BPCRM,30,D, 1,1,DATCNT,YES
7370  025526  014052                                    .WORD   T$CODE
7371  025530  026431                                    .WORD   BPCRM
7372  025532  177777                                    .WORD    1
7373  025534  000001                                    .WORD   T$LOLIM
7374  025536  004000                                    .WORD   T$HILIM
7375  025540                    GPRMD   NUMBM,32,D, 1,1,77777,YES
7376  025540  015052                                    .WORD   T$CODE
7377  025542  026443                                    .WORD   NUMBM
7378  025544  177777                                    .WORD    1
7379  025546  000001                                    .WORD   T$LOLIM
7380  025550  077777                                    .WORD   T$HILIM
7381  025552                    GPRMD   PATTM,34,D,17,0,10,YES
7382  025552  016052                                    .WORD   T$CODE
7383  025554  026463                                    .WORD   PATTM
7384  025556  000017                                    .WORD   17
7385  025560  000000                                    .WORD   T$LOLIM
7386  025562  000010                                    .WORD   T$HILIM
7387  025564                    GPRMD   CMD4M,36,D,37,1,33,YES
7388  025564  017052                                    .WORD   T$CODE
```

```
7389  025566  026504                                                    .WORD   CMD4M
7390  025570  000037                                                    .WORD   37
7391  025572  000001                                                    .WORD   T$LOLIM
7392  025574  000033                                                    .WORD   T$HILIM
7393  025576                        GPRMD   BPCRM,40,D,-1,1,DATCNT,YES
7394  025576  020052                                                    .WORD   T$CODE
7395  025600  026431                                                    .WORD   BPCRM
7396  025602  177777                                                    .WORD   -1
7397  025604  000001                                                    .WORD   T$LOLIM
7398  025606  004000                                                    .WORD   T$HILIM
7399  025610                        GPRMD   NUMBM,42,D, 1,1,77777,YES
7400  025610  021052                                                    .WORD   T$CODE
7401  025612  026443                                                    .WORD   NUMBM
7402  025614  177777                                                    .WORD   -1
7403  025616  000001                                                    .WORD   T$LOLIM
7404  025620  077777                                                    .WORD   T$HILIM
7405  025622                        GPRMD   PATTM,44,D,17,0,10,YES
7406  025622  022052                                                    .WORD   T$CODE
7407  025624  026463                                                    .WORD   PATTM
7408  025626  000017                                                    .WORD   17
7409  025630  000000                                                    .WORD   T$LOLIM
7410  025632  000010                                                    .WORD   T$HILIM
7411  025634                        GPRMD   CMD5M,46,D,37,1,33,YES
7412  025634  023052                                                    .WORD   T$CODE
7413  025636  026512                                                    .WORD   CMD5M
7414  025640  000037                                                    .WORD   37
7415  025642  000001                                                    .WORD   T$LOLIM
7416  025644  000033                                                    .WORD   T$HILIM
7417  025646                        GPRMD   BPCRM,50,D, 1,1,DATCNT,YES
7418  025646  024052                                                    .WORD   T$CODE
7419  025650  026431                                                    .WORD   BPCRM
7420  025652  177777                                                    .WORD   -1
7421  025654  000001                                                    .WORD   T$LOLIM
7422  025656  004000                                                    .WORD   T$HILIM
7423  025660                        GPRMD   NUMBM,52,D, 1,1,77777,YES
7424  025660  025052                                                    .WORD   T$CODE
7425  025662  026443                                                    .WORD   NUMBM
7426  025664  177777                                                    .WORD   -1
7427  025666  000001                                                    .WORD   T$LOLIM
7428  025670  077777                                                    .WORD   T$HILIM
7429  025672                        GPRMD   PATTM,54,D,17,0,10,YES
7430  025672  026052                                                    .WORD   T$CODE
7431  025674  026463                                                    .WORD   PATTM
7432  025676  000017                                                    .WORD   17
7433  025700  000000                                                    .WORD   T$LOLIM
7434  025702  000010                                                    .WORD   T$HILIM
7435  025704                        XFER    ENDSP2
7436  025704  002004                                                    .WORD   T$CODE
7437  025706               ENDSP1:  XFER    ENDSP
7438  025706  075004                                                    .WORD   T$CODE
7439  025710               ENDSP2:  GPRMD   CMD6M,56,D,37,1,33,YES
7440  025710  027052                                                    .WORD   T$CODE
7441  025712  026520                                                    .WORD   CMD6M
7442  025714  000037                                                    .WORD   37
7443  025716  000001                                                    .WORD   T$LOLIM
7444  025720  000033                                                    .WORD   T$HILIM
```

```
7445  025722                    GPRMD   BPCRM,60,D, 1,1,DATCNT,YES
7446  025722  030052                                              .WORD   T$CODE
7447  025724  026431                                              .WORD   BPCRM
7448  025726  177777                                              .WORD    1
7449  025730  000001                                              .WORD   T$LOLIM
7450  025732  004000                                              .WORD   T$HILIM
7451  025734                    GPRMD   NUMBM,62,D,-1,1,77777,YES
7452  025734  031052                                              .WORD   T$CODE
7453  025736  026443                                              .WORD   NUMBM
7454  025740  177777                                              .WORD    1
7455  025742  000001                                              .WORD   T$LOLIM
7456  025744  077777                                              .WORD   T$HILIM
7457  025746                    GPRMD   PATTM,64,D,17,0,10,YES
7458  025746  032052                                              .WORD   T$CODE
7459  025750  026463                                              .WORD   PATTM
7460  025752  000017                                              .WORD   17
7461  025754  000000                                              .WORD   T$LOLIM
7462  025756  000010                                              .WORD   T$HILIM
7463  025760                    GPRMD   CMD7M,66,D,37,1,33,YES
7464  025760  033052                                              .WORD   T$CODE
7465  025762  026526                                              .WORD   CMD7M
7466  025764  000037                                              .WORD   37
7467  025766  000001                                              .WORD   T$LOLIM
7468  025770  000033                                              .WORD   T$HILIM
7469  025772                    GPRMD   BPCRM,70,D, 1,1,DATCNT,YES
7470  025772  034052                                              .WORD   T$CODE
7471  025774  026431                                              .WORD   BPCRM
7472  025776  177777                                              .WORD   -1
7473  026000  000001                                              .WORD   T$LOLIM
7474  026002  004000                                              .WORD   T$HILIM
7475  026004                    GPRMD   NUMBM,72,D, 1,1,77777,YES
7476  026004  035052                                              .WORD   T$CODE
7477  026006  026443                                              .WORD   NUMBM
7478  026010  177777                                              .WORD   -1
7479  026012  000001                                              .WORD   T$LOLIM
7480  026014  077777                                              .WORD   T$HILIM
7481  026016                    GPRMD   PATTM,74,D,17,0,10,YES
7482  026016  036052                                              .WORD   T$CODE
7483  026020  026463                                              .WORD   PATTM
7484  026022  000017                                              .WORD   17
7485  026024  000000                                              .WORD   T$LOLIM
7486  026026  000010                                              .WORD   T$HILIM
7487  026030                    GPRMD   CMD8M,76,D,37,1,33,YES
7488  026030  037052                                              .WORD   T$CODE
7489  026032  026534                                              .WORD   CMD8M
7490  026034  000037                                              .WORD   37
7491  026036  000001                                              .WORD   T$LOLIM
7492  026040  000033                                              .WORD   T$HILIM
7493  026042                    GPRMD   BPCRM,100,D,-1,1,DATCNT,YES
7494  026042  040052                                              .WORD   T$CODE
7495  026044  026431                                              .WORD   BPCRM
7496  026046  177777                                              .WORD   -1
7497  026050  000001                                              .WORD   T$LOLIM
7498  026052  004000                                              .WORD   T$HILIM
7499  026054                    GPRMD   NUMBM,102,D, 1,1,77777,YES
7500  026054  041052                                              .WORD   T$CODE
```

```
 7501  026056  026443                                                    .WORD   NUMBM
 7502  026060  177777                                                    .WORD   -1
 7503  026062  000001                                                    .WORD   T$LOLIM
 7504  026064  077777                                                    .WORD   T$HILIM
 7505  026066                          GPRMD   PATTM,104,D,17,0,10,YES
 7506  026066  042052                                                    .WORD   T$CODE
 7507  026070  026463                                                    .WORD   PATTM
 7508  026072  000017                                                    .WORD   17
 7509  026074  000000                                                    .WORD   T$LOLIM
 7510  026076  000010                                                    .WORD   T$HILIM
 7511  026100                  ENDSP:
 7512  026100                          XFER    JMPMSG
 7513  026100  176004                                                    .WORD   T$CODE
```

```
7514
7515
                                                        .NLIST  BEX
026102  046103  040505  020122          CLRM:   .ASCIZ  /CLEAR COUNTERS/
026121     122  051505  052105          RRVM:   .ASCIZ  /RESET RANDOM VARIABLES/
026150  040510  052114  040440          HAEM:   .ASCIZ  /HALT AFTER EACH CMD/
026174  051120  047111  020124          RCVERM: .ASCIZ  /PRINT RECOVERABLE ERRORS/
026225     111  044116  041111          IRECM:  .ASCIZ  /INHIBIT RECOVERY/
026246  040502  020104  040524          BADTM:  .ASCIZ  /BAD TAPE SPOT DETECTION/
026276  044504  040523  046102          DINTM:  .ASCIZ  /DISABLE INTERRUPTS/
026321     111  044116  041111          IREM:   .ASCIZ  /INHIBIT RFC ERROR REPORT/
026352  044103  047101  042507          CHGM:   .ASCIZ  /CHANGE CMD SEQUENCE/
026376  044103  051101  041501          CHARM:  .ASCIZ  /CHARACTERISTICS CODE/
026423     103  042115  031057          CMD2M:  .ASCIZ  "CMD/2"
026431     102  043122  041440          BPCRM:  .ASCIZ  /BRF COUNT/
026443     043  047440  020106          NUMBM:  .ASCIZ  /# OF OPERATIONS/
026463     120  052101  042524          PATTM:  .ASCIZ  /PATTERN/
                                                        .LIST   BEX
7516    026474                                          .EVEN
7517
7518    026474                          JMPMSG:
7519    026474                                  EXIT SFT
7520    026474  023004                                                          .WORD   T$CODE
7521
7522                                                    .NLIST  BEX
026476  046503  027504  000063          CMD3M:  .ASCIZ  "CMD/3"
026504  046503  027504  000064          CMD4M:  .ASCIZ  "CMD/4"
026512  046503  027504  000065          CMD5M:  .ASCIZ  "CMD/5"
026520  046503  027504  000066          CMD6M:  .ASCIZ  "CMD/6"
026526  046503  027504  000067          CMD7M:  .ASCIZ  "CMD/7"
026534  046503  027504  000070          CMD8M:  .ASCIZ  "CMD/8"
                                                        .LIST   BEX
7523                                                    .EVEN
7524
7525    026542                                  ENDSFT
7526                                                                            .EVEN
7527    026542                          L10041::
7528
7529                                    ;***********************************************************************
7530                                    ;***********************************************************************
7531                                    ;       PATCH AREA
7532
7533    026542  000100                  PATCH:: .BLKW   64.
7534                                    ;***********************************************************************
7535                                    ;***********************************************************************
7536
7537    026742                                  LAST:TAD
7538                                                                            .EVEN
7539    026742  026756                                                          .WORD   T$FREE
7540    026744  000004                                                          .WORD   T$SIZE
7541    026746                          L$LAST::
7542    026746                                  ENDMOD
```

```
7543                                    .SBTTL  HARD CODED P TBL
7544
7545                                    ;++
7546                                    ;DIAG IS PRE PARAMETERIZED PER TBL
7547                                    ;--
7548
7549  026746                                    BGNSETUP 1
7550  026746                                      BGNPTAB
7551  026746  000000                                                    .WORD   0
7552  026750  000002                                                    .WORD   L10044 .
7553  026752                                    L10042:
7554  026752  172522                                  172522
7555  026754  000224                                  224
7556  026756                                    ENDPTAB
7557  026756                                    L10044:
7558  026756                                    ENDSETUP
7559
7560          000001                          .END
```

```
ACK.C = 100000 G
ADR   = 000020 G
ALLEOT  003450 G
ASSEMB= 000010
ATTNM   004335 G
AUDRPM  004645 G
AUTODM  021522
BADTM   026246
BADTSW  002206 G
BFSEQ   022676
BFSEQ0  022722
BFSEQ1  022774
BFSEQ2  023006
BFSEQ3  023100
BFSEQ4  023152
BFSEQ5  023214
BFSEQ6  023266
BFSEQ7  023320
BFSEQ8  023352
BFSEQ9  023404
BFSE10  023426
BGNFLG= 003404
BINC    014366
BIT0  = 000001 G
BIT00 = 000001 G
BIT01 = 000002 G
BIT02 = 000004 G
BIT03 = 000010 G
BIT04 = 000020 G
BIT05 = 000040 G
BIT06 = 000100 G
BIT07 = 000200 G
BIT08 = 000400 G
BIT09 = 001000 G
BIT1  = 000002 G
BIT10 = 002000 G
BIT11 = 004000 G
BIT12 = 010000 G
BIT13 = 020000 G
BIT14 = 040000 G
BIT15 = 100000 G
BIT2  = 000004 G
BIT3  = 000010 G
BIT4  = 000020 G
BIT5  = 000040 G
BIT6  = 000100 G
BIT7  = 000200 G
BIT8  = 000400 G
BIT9  = 001000 G
BOE   = 000400 G
BORERS  013576 G
BPCRM   026431
BRCPK   002324 G
BRFCNT  003344 G
BRF.C = 004000 G
BTADDR  002544 G
BTMSG1  013372
BTMSG2  013457
BTMSG3  013527
BTPT    003436 G
BTRPT   016544
BTO     002774 G
BT1     003046 G
BT2     003120 G
BT3     003172 G
CHAR    002216 G
CHARM   026376
CHGFLG  002212 G
CHGM    026352
CHKERR  011456 G
CH.EAI= 000040 G
CH.ERI= 000020 G
CH.ESS= 000200 G
CKDATA  015036 G
CKDCNT  015446
CKDFF   015450
CKHAE   016060 G
CKHRTN  016146
CLN   = 101012 G
CLRERR  011154 G
CLRFLG  002202 G
CLRM    026102
CMDAC   007344 G
CMDASC  003650 G
CMDD    002220 G
CMDLG   003354 G
CMDPKM  004102 G
CMDPKT  002310 G
CMDSAV  003350 G
CMDSEQ  003460 G
CMDSE2  003470 G
CMDTBL  003562 G
CMDWRD  003346 G
CMD.C0= 000001 G
CMD.C1= 000002 G
CMD.C2= 000004 G
CMD.C3= 000010 G
CMD.C4= 000020 G
CMD2M   026423
CMD3M   026476
CMD4M   026504
CMD5M   026512
CMD6M   026520
CMD7M   026526
CMD8M   026534
CNTBGN- 002554
CNTEND= 003324
CNTLEN= 000550 G
CODELM  003772 G
COUNTE= 050404
CP.ADH= 000004 G
CP.ADL= 000002 G
CP.CMD= 000000 G
CP.CNT= 000006 G
CRLF    005213 G
CRLFSP  005216 G
CTCC    003376 G
CVC.C = 040000 G
C$AU  = 000052
C$AUTO= 000061
C$BRK = 000022
C$BSEG= 000004
C$BSUB= 000002
C$CEFG= 000045
C$CLCK= 000062
C$CLEA= 000012
C$CLOS= 000035
C$CLP1= 000006
C$CVEC= 000036
C$DCLN= 000044
C$DODU= 000051
C$DRPT= 000024
C$DU  = 000053
C$EDIT= 000003
C$ERDF= 000055
C$ERHR= 000056
C$ERRO= 000060
C$ERSF= 000054
C$ERSO= 000057
C$ESCA= 000010
C$ESEG= 000005
C$ESUB= 000003
C$ETST= 000001
C$EXIT= 000032
C$GETB= 000026
C$GETW= 000027
C$GMAN= 000043
C$GPHR= 000042
C$GPLO= 000030
C$GPRI= 000040
C$INIT= 000011
C$INLP= 000020
C$MANI  000050
C$MEM = 000031
C$MSG = 000023
C$OPEN= 000034
C$PNTB= 000014
C$PNTF= 000017
C$PNTS= 000016
C$PNTX= 000015
C$QIO = 000377
C$RDBU= 000007
C$REFG= 000047
C$RESE= 000033
C$REVI= 000003
C$RFLA= 000021
C$RPT = 000025
C$SEFG= 000046
C$SPRI= 000041
C$SVEC= 000037
C$TPRI= 000013
DATARD  003336 G
DATAWT  003334 G
DATCNT= 004000 G
DEVTBL  002532 G
DFPTBL  002174 G
DFTSCH= 000040 G
DIA   = 100006 G
DIABLK= 003334 G
DIACNT= 000020 G
DIAGMC= 000000
DINT    002210 G
DINTM   026276
DLY   = 000020 G
DLY.C = 000020 G
DRI   = 100013 G
DROPDM  004616 G
DROPED  003446 G
DROPN   015774
DROPU   015554 G
DROPUA  015704
DRORTN  015762
DTAERM  005224 G
DTAER2  004677 G
DTAER3  004746 G
DTAER4  005010 G
DTAER5  005031 G
EF.CON= 000036 G
EF.NEW= 000035 G
EF.PWR= 000034 G
EF.RES= 000037 G
EF.STA= 000040 G
EINC    014374
END   = 177777 G
ENDERF= 003416
ENDFLG= 003452
ENDSP   026100
ENDSP1  025706
ENDSP2  025710
EOTFLG  003426 G
ERCVER  002205 G
ERLOG   003412 G
ERRREC  003415 G
ERS   = 100411 G
ERSFLG  003451 G
EVL   = 000004 G
EXALL   006510 G
EXARTN  007002
EXCRTN  010634
EXCUTE  010326 G
EXPBOT  003440 G
EXSUB   007004 G
E$END = 002100
E$LOAD= 000035
FATSM   004373 G
FIRSTU  015452 G
FMT.C0= 000040 G
FMT.C1= 000100 G
FTLCNT  003314 G
FUNRM   004353 G
F$AU  = 000015
F$AUTO= 000020
F$BGN = 000040
F$CLEA= 000007
F$DU  = 000016
F$END = 000041
F$HARD= 000004
F$HW  = 000013
F$INIT= 000006
F$JMP = 000050
F$MOD = 000000
F$MSG = 000011
F$PROT= 000021
F$PWR = 000017
F$RPT = 000012
F$SEG = 000003
F$SOFT= 000005
F$SRV = 000010
F$SUB = 000002
F$SW  = 000014
F$TEST= 000001
GCMDA   007416 G
GENPAT  010030 G
GES   = 100017 G
GETSTM  005157 G
GIT     010322
GOWAIT  010636 G
GSCPK   002320 G
G$CNTO= 000200
G$DELM= 000372
G$DISP= 000003
G$EXCP= 000400
G$HILI= 000002
G$LOLI= 000001
G$NO  = 000000
```

```
G$OFFS= 000400          JMP    = 000040 G      L$SPC   002056 G      MSGPKT  002334 G      PRIO1  = 000040 G
G$OFSI= 000376          JMPMSG  026474         L$SPCP  002020 G      MSGPK0  002352 G      PRIO2  = 000100 G
G$PRMA= 000001          JMP.C  = 000040 G      L$SPTP  002024 G      MSGPK1  002370 G      PRIO3  = 000140 G
G$PRMD= 000002          J$JMP  = 000167        L$STA   002030 G      MSGPK2  002406 G      PRIO4  = 000200 G
G$PRML= 000000          LENMSK  003356 G       L$SW    002202 G      MSGPK3  002424 G      PRIO5  = 000240 G
G$RADA= 000140          LOE    = 040000 G      L$TEST  002114 G      MS.RFC= 000004 G      PRIO6  = 000300 G
G$RADB= 000000          LOG     014102 G       L$TIML  002014 G      MS.XS0= 000006 G      PRIO7  = 000340 G
G$RADD= 000040          LOT    = 000010 G      L$UNIT  002012 G      MS.XS1= 000010 G      PRXST   015776 G
G$RADL= 000120          L$ACP   002110 G       L10000  002200        MS.XS2= 000012 G      PTCMDS  025134
G$RADO= 000020          L$APT   002036 G       L10001  002310        MS.XS3= 000014 G      PWRFLG  003453 G
G$XFER= 000004          L$AU    021770 G       L10002  005370        NCMD.C= 177740 G      RANB    003360 G
G$YES = 000010          L$AUT   002070 G       L10003  006314        NCNT    003340 G      RANBC  = 153624 G
HAE     002204 G        L$AUTO  021232 G       L10004  006322        NCNT1   003342 G      RANCMD  023732
HAEM    026150          L$CCP   002106 G       L10005  006330        NEXTSP  025406        RANDOM  003441 G
HALTM   004042 G        L$CLEA  021654 G       L10006  006336        NEXTU   015520 G      RANP   = 000007 G
HELP   = 000000         L$CO    002032 G       L10007  006344        NINUSE= 177774 G      RANRD   023772
HOE    = 100000 G       L$DEPO  002011 G       L10010  017674        NOINTM  004421 G      RANS    003362 G
HRDCNT  003304 G        L$DESC  002136 G       L10012  021230        NRDYM   021616        RANSC  = 032561 G
IBE    = 010000 G       L$DESP  002076 G       L10013  021520        NSSRM   004271 G      RANW    024072
IDU    = 000040 G       L$DEVP  002060 G       L10014  021714        NUMBM   026443        RANWR   024046
IER    = 020000 G       L$DISP  002124 G       L10015  021766        NURTY1  005073 G      RANWV   024060
IE.C   = 000200 G       L$DLY   002116 G       L10016  022062        OFLINM  005127 G      RCVERM  026174
INIT10  017704          L$DTP   002040 G       L10017  023450        ONEFIL= 000001        RDF    = 104001 G
INIT15  020132          L$DTYP  002034 G       L10020  022216        OPFLAG  003456 G      RDR    = 104401 G
INIT16  020152          L$DU    021716 G       L10021  022242        OPP.C  = 020000 G      RECCNT  003324 G
INTFLG  003416 G        L$DUT   002072 G       L10022  022262        O$APTS= 000000        RECLOG  003411 G
INTPRI= 000340 G        L$DVTY  002164 G       L10023  022302        O$AU   = 000001        RECRED  006312
IRE     003445 G        L$EF    002052 G       L10024  022322        O$BGNR= 000001        RECTAP  006346 G
IREC    002211 G        L$ENVI  002044 G       L10025  022342        O$BGNS= 000001        RECUD   011310 G
IRECM   026225          L$ETP   002102 G       L10026  022362        O$DU   = 000001        REPEAT= 050224
IREM    026321          L$EXP1  002046 G       L10027  022402        O$ERRT= 000000        RERM    004550 G
ISR    = 000100 G       L$EXP4  002064 G       L10030  022422        O$GNSW= 000001        RETRY  = 050222
IXE    = 004000 G       L$EXP5  002066 G       L10031  022442        O$POIN= 000001        RETRYC  003404 G
I$AU   = 000041          L$HARD  025266 G       L10032  022500        O$SETU= 000001        REWRT   013752
I$AUTO= 000041          L$HIME  002120 G       L10033  022664        PASCNT  003254 G      RFBC    002654 G
I$CLN  = 000041          L$HPCP  002016 G       L10034  024124        PATCH   026542 G      RFCERM  004254 G
I$DU   = 000041          L$HPTP  002022 G       L10035  024270        PATERN  003374 G      RFREC   002754 G
I$HRD  = 000041          L$HW    002174 G       L10036  024422        PATR0   010114 G      RFUNR   002764 G
I$INIT= 000041          L$ICP   002104 G       L10037  025262        PATR1   010152 G      RLEXM   004310 G
I$MOD  - 000041          L$INIT  017704 G       L10040  025336        PATR2   010172 G      RNF    = 125401 G
I$MSG  = 000041          L$LADP  002026 G       L10041  026542        PATR3   010202 G      RNOPSC= 177740 G
I$PROT= 000040          L$LAST  026746 G       L10042  026752        PATR4   010226 G      RNR    = 105401 G
I$PTAB= 000041          L$LOAD  002100 G       L10044  026756        PATR5   010240 G      RNTM    004504 G
I$PWR  = 000041          L$LUN   002074 G       MBR    = 100012 G     PATR6   010252 G      RPF    = 105001 G
I$RPT  = 000041          L$MREV  002050 G       MEMOM   021126        PATR7   010272 G      RPR    = 125001 G
I$SEG  = 000041          L$NAME  002000 G       MISCFG  003455 G      PATR8   010324 G      RPTCNT  003406 G
I$SETU= 000041          L$PRIO  002042 G       MOD.C0= 000400 G      PATTBL  010072        RPTFLG  003443 G
I$SFT  = 000041          L$PROT  017676 G       MOD.C1= 001000 G      PATTM   026463        RPT1A   017012
I$SRV  = 000041          L$PRT   002112 G       MOD.C2= 002000 G      PCMDWD  003352 G      RPT1B   017067
I$SUB  = 000041          L$REPP  002062 G       MOD.C3= 004000 G      PIRE    002214 G      RPT1C   017140
I$TST  = 000041          L$REV   002010 G       MOVMSG  011224 G      PNT    = 001000 G      RPT1D   017211
JLOC    003372 G        L$RPT   016150 G       MSGCNT= 000016 G      PRI    = 002000 G      RPT1E   017437
JLOOP   003370 G        L$SOFT  025340 G       MSGPKA  002502 G      PRIOO  = 000000 G      RPT1F   017315
```

```
RPT1G   017366          TCC5    012552 G        T$PTAB= 010043          URERM   004572 G        $F$LOO= 000200
RPT1I   017563          TCC6    012650 G        T$PTHV= 000001          VFEXC   014504 G        $F$NAM= 000160
RPT1J   017467          TCC7    012700 G        T$PTNU= 000001          VFISU   014750 G        $F$NO = 000403
RPT1K   017554          TC2RTN  012062          T$SAVL= 177777          VFYCNT  003274 G        $F$OR = 000320
RRANV   002203 G        TIME1   003364 G        T$SEGL= 177777          VFYDAT  014402 G        $F$RTN= 000300
RRBC    002614 G        TIME2   003366 G        T$SIZE= 000004          VFYFLG  003442 G        $F$SEL= 000140
RRECL = 000020 G        TOERM   004207 G        T$SUBN= 000000          VFY.C = 000100 G        $F$THE  000330
RREC    002734 G        TOOMM   004460 G        T$TAGL= 177777          WLKZRO  010206          $F$TRU= 000404
RRUNR   002744 G        TRAPD4  003454 G        T$TAGN= 010045          WRBC    002554 G        $F$UNT= 000130
RRVM    026121          TRAP4   021646 G        T$TEMP= 000000          WRECL = 000020 G        $F$WHI= 000120
RTLE    012716 G        TSAM    004436 G        T$TEST= 000005          WRR   = 105005 G        $F$YES= 000402
RTLRTN  013042          TSBA  = 002452 G        T$TSTM= 177777          WRREC   002714 G        $IFLEV= 177777
RWCPK   002330 G        TSC.FC= 177717 G        T$TSTS= 000001          WRT   = 104005 G        $ISK0 = 000001
RWD   = 102010 G        TSC.TC= 177761 G        T$$AU = 010016          WRTY    013044 G        $ISK1 = 000001
RWERR   003413 G        TSDB    002452 G        T$$AUT= 010013          WRTYCT  003244 G        $ISK2 = 000001
R5SAVE  003400 G        TSSR    002462 G        T$$CLE= 010014          WRTYER  003410 G        $ISK3 = 000001
SCCNT   003264 G        TSSREG  003402 G        T$$DAT= 010044          WRTYFG  003407 G        $ISK4 = 000001
SCERM   004230 G        TSVCT   002472 G        T$$DU = 010015          WRUNR   002724 G        $ISK5 = 000001
SCH   = 140004 G        TS.A16= 000400 G        T$$HAR= 010040          WSSR    011170 G        $ISK6 = 000001
SCHBK   002442 G        TS.A17= 001000 G        T$$HW = 010000          WTM   = 100011 G        $LOCTA= 177777
SCHCNT= 000010 G        TS.NBA= 002000 G        T$$INI= 010012          WTR   = 101011 G        $LSTCN= 177777
SEQEND  003550 G        TS.NXM= 004000 G        T$$MSG= 010003          WTV   = 104105 G        $LSTIN= 000001
SETCH   006444 G        TS.OFL= 000100 G        T$$PC = 000001          WTVERM  004164 G        $LSTST= 177777
SETRW   006470 G        TS.RMR= 010000 G        T$$PRO= 010011          WTYBRF  013370          $LSTTA= 000001
SETUP   007452 G        TS.SC = 100000 G        T$$PTA= 010043          WTYCMD  013364          $MCALL= 000000
SFF   = 105010 G        TS.SPE= 020000 G        T$$RPT= 010010          WTYWRD  013366          $NESTL= 177777
SFPTBL  002202 G        TS.SSR= 000200 G        T$$SOF= 010041          X$ALWA= 000000          $NSK0 = 000120
SFR   = 105410 G        TS.UPE= 040000 G        T$$SRV= 010007          X$FALS= 000040          $NSK1 = 000120
SRF   = 104010 G        TS4ADR  025312          T$$SUB= 010033          X$OFFS= 000400          $NSK2 = 000110
SRR   = 104410 G        TS4CL   002522 G        T$$SW = 010001          X$TRUE= 000020          $NSK3 = 000110
STAERM  005372 G        TS4INT  002512 G        T$$TES= 010037          XO.BOT= 000002 G        $NSK4 = 000110
STAER1  005704          TS4IN0  006316 G        T1      022064 G        XO.EOT= 000001 G        $NSK5 = 000110
STAER2  006062          TS4IN1  006324 G        T1SWB   003447 G        XO.LET= 020000 G        $NSK6 = 000110
STAER3  006141          TS4IN2  006332 G        T1.1    022074          XO.ONL= 000100 G        $SAVLE= 177777
STAER4  006177          TS4IN3  006340 G        T1.10   022424          XO.RLL= 010000 G        $SSK0 = 050452
STAER5  006217          TS4VCT  025327          T1.11   022444          XO.RLS= 040000 G        $TAGLE= 177777
STAER6  006026          T$ARGC= 000003          T1.12   022530          XO.TMK= 100000 G        $TAGNU= 050470
STAER7  005776          T$CODE= 023004          T1.2    022220          X2.OPM= 100000 G        $TEMP = 000402
STAFLG  003452 G        T$ERRN= 000001          T1.3    022244          X3.DCK= 000010 G        $TSK0 = 050451
SVCGBL= 000000          T$EXCP= 000000          T1.4    022264          X3.RNY= 157400 G        $TSK1 = 050452
SVCINS= 000001          T$FLAG= 000041          T1.5    022304          ZROPAT  010156          $TSK2 = 050453
SVCSUB= 000000          T$FREE= 026756          T1.6    022324          $BGNLE= 177777          $TSK3 = 050454
SVCTAG= 000000          T$GMAN= 000000          T1.7    022344          $ERFLG= 000400          $TSK4 = 050467
SVCTST= 000000          T$HILT= 000010          T1.8    022364          $F$AND= 000310          $TSK5 = 050466
SWBFLG  003444 G        T$LAST= 000001          T1.9    022404          $F$BAD= 000401          $TSK6 = 050462
SWB.C = 010000 G        T$LOLI= 000000          T2      023452 G        $F$BLA= 000170          $TSK7 = 050464
S$LSYM= 010000          T$LSYM= 010000          T3      024126 G        $F$CAS= 000150          $$ARGC= 000000
TCCRA   011700          T$LTNO= 000005          T4      024272 G        $F$DEC= 000220          $$BYTE= 000403
TCC0    011720 G        T$NEST= 177777          T5      024424 G        $F$DO = 000340          $$CASE= 000000
TCC1    011736 G        T$NS0 = 000000          T5WEOT  025156          $F$FAL= 000405          $$DST = 000000
TCC2    011754 G        T$NS1 = 000005          UAM   = 000200 G        $F$GOO= 000400          $$ELOC= 000402
TCC3    012064 G        T$NS2 = 000002          UNL   = 100412 G        $F$IF = 000110          $$ERFL= 000000
TCC4    012102 G        T$PCNT= 000000          UNREC   003414 G        $F$INC= 000210          $$FLAG= 000001
```

```
PARAMETER CODING        MACY11 30(1046)  06 APR 84  08:51  PAGE 184
CZTSHD.P11     06 APR 84 08:49          SYMBOL TABLE                                    SEQ 0181

$$FROM= 000000          $$REG = 177777          $$RTN2= 000000          $$TGS1= 000000          $$$TAG= 050000
$$LOC = 025072          $$RETU= 000000          $$SRC = 000000          $$TGS2  000000          .      = 026756
$$LOCN  000000          $$RTN1= 000000          $$TGSV= 000000          $$TO   = 000000

. ABS.  026756      000


 ERRORS DETECTED:  0

 CZTSHD,CZTSHD/SOL/EQ:ONEFILE=SVC.SML,SPMAC.SML,CZTSHD.P11
 RUN-TIME: 132 138 .8 SECONDS
 RUN TIME RATIO: 347/271=1.2
 CORE USED:  31K  (62 PAGES)
```